AD-A010 834

EVALUATION OF AN INTEGRATED ELECTRONIC INSTRUMENT DISPLAY FOR HELICOPTER HOVER OPERATIONS USING A SIX-DEGREE-OF-FREEDOM FIXED-BASE SIMULATION

Larry Richard Ammerman

Naval Postgraduate School
Monterey, California

March 1975

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

EVALUATION OF AN INTEGRATED ELECTRONIC INSTRUMENT DISPLAY FOR HELICOPTER HOVER OPERATIONS USING A SIX-DEGREE-OF-FREEDOM FIXED-BASE SIMULATION

by

Larry Richard Ammerman

Thesis Advisor: D. M. Layton

March 1975

95

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Evaluation of an Integrated Electronic Instrument Display for Helicopter Hover Operations using a Six-Degree-of-Freedom Fixed-Base Simulation		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; March 1975
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Larry Richard Ammerman		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1975
		13. NUMBER OF PAGES 95
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

instrument flight
helicopter
integrated display

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report discusses the development and evaluation of an integrated electronic instrument display designed to help alleviate pilot work load and improve aircraft control during the precision hover task while flying solely by reference to instruments. The evaluation utilizes a hybrid computer system to implement a six-degree-of-freedom fixed-base simulation of the SH-2 helicopter and a graphics processor to generate the integrated instrument display. Evaluation pilots were asked to rate the integrated display against

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601 1

PRICES SUBJECT TO CHANGE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20.
conventional flight instruments after flying a simulated night over-water rescue mission. The evaluation revealed that the simulated aircraft dynamics were susceptible to pilot induced oscillations in a hover and, therefore, unsatisfactory for use as an evaluation tool. In general, the evaluation pilots considered the integrated display preferable to conventional cockpit instruments; however, further study is recommended since meaningful quantitative data were not obtained.

Evaluation of an Integrated Electronic Instrument
Display for Helicopter Hover Operations using
a Six-Degree-of-Freedom Fixed-Base Simulation

by

Larry Richard Ammerman
Lieutenant, United States Navy
B.S., The Pennsylvania State University, 1968

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1975

Author ______________________

Approved by: ______________________ Thesis Advisor

______________________ Chairman, Department of Aeronautics

______________________ Academic Dean

ABSTRACT

This report discusses the development and evaluation of an integrated electronic instrument display designed to help alleviate pilot work load and improve aircraft control during the precision hover task while flying solely by reference to instruments. The evaluation utilizes a hybrid computer system to implement a six-degree-of-freedom fixed-base simulation of the SH-2F helicopter and a graphics processor to generate the integrated instrument display. Evaluation pilots were asked to rate the integrated display against conventional flight instruments after flying a simulated night over-water rescue mission. The evaluation revealed that the simulated aircraft dynamics were susceptible to pilot induced oscillations in a hover and, therefore, unsatisfactory for use as an evaluation tool. In general, the evaluation pilots considered the integrated display preferable to conventional cockpit instruments: however, further study is recommended since meaningful quantitative data were not obtained.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

g	Acceleration of gravity, ft/sec^2
I_{xx}, I_{yy}, I_{zz}	Moment of inertia about the x, y, z body-fixed axes respectively, slug-ft^2
I_{xz}	Product of inertia, $\int xz\, dm$, slug-ft^2
L, M, N	Aerodynamic moments about the x, y, z body-fixed axes respectively, ft-lbs
m	Mass of the helicopter, slugs
p, q, r	Angular rates about the x, y, z body-fixed axes respectively, radians/sec
u, v, w	Velocities along the x, y, z body-fixed axes respectively, ft/sec
V_x, V_y, V_z	Velocities along the X, Y, Z axes of the inertial reference frame, ft/sec
X_A, Y_A, Z_A	Aerodynamic forces along the x, y, z body-fixed axes respectively, lbs
X_E, Y_E, Z_E	Coordinate position of helicopter in the inertial reference frame, ft
ΔA_{1c}, ΔB_{1c}	Change in lateral and longitudinal cyclic pitch respectively from reference values, radians
$\Delta\theta_c$	Change in main rotor collective pitch from reference value, radians
$\Delta\theta_r$	Change in tail rotor collective pitch from reference value, radians

Subscripts

p, q, r, u, v, w A_{1c}, B_{1c}, θ_c, θ_r	Used with X, Y, Z, L, M, N to indicate the partial derivative of an aerodynamic force or moment with respect to p, q, r, u, v, w, A_{1c}, B_{1c}, θ_c, θ_r respectively

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to Mr. Robert Limes, Mr. William Thomas, and Mr. Al Wong of the Computer Science Laboratory staff for their patient and helpful assistance in programming the simulation, and to Associate Professor Donald M. Layton for his advice and also his technical assistance on the flight trainer. The author adds a special thanks to his wife, Karen, for her assistance and understanding during the project.

I. INTRODUCTION

Almost without exception, knowledgeable helicopter pilots will agree that a lengthy precision hover conducted solely by reference to instruments is one of the most demanding of all helicopter piloting skills. The term "knowledgeable" is used because the vast majority of helicopter pilots are not sufficiently familiar with the instrument hover task to appreciate its difficulty. Most missions of which the helicopter is capable, including both civil and military, do not require a lengthy precision hover under instrument flight conditions. Two principle exceptions to this situation are the Navy anti-submarine warfare (ASW) mission and the Navy and Coast Guard requirement for all-weather, day-night, over-water rescue capability. At a recent Helicopter Instrument Flight Conference hosted by the U.S. Army Aviation Systems Command [Ref. 1] only the U.S. Navy identified the need for increased capability to conduct precision hover operations under instrument flight conditions. A Navy representative to the conference described night over-water flight conditions as an "ink bottle" flight environment requiring the pilot to be on the gages 100% of the time.

Although the Navy has attempted to reduce the pilot work load required for the precision hover task by the use of automatic flight control systems (AFCS), it has been the author's experience that present generation AFCS reliability leaves much to be desired. Mission completion has often been the result of skillful piloting requiring the use of conventional flight instruments. Future generation AFCS reliability will probably improve but the availability of a good

secondary capability should not be ignored in view of the critical requirements for both flight safety and mission completion. It is felt that such a capability can be achieved by providing the pilot with an improved flight instrument display. Such a display should provide the pilot with conventional flight instrument information in a compact, easily recognizable form. The compactness should reduce scan time and improve control and the easily recognizable form should allow a smooth transition from conventional flight instruments to the improved display. It was proposed that such a display in the form of an integrated electronic instrument display would satisfy the above requirements.

An integrated instrument display was developed using the Scientific Data Systems model 9300 digital computer and the Adage AGT-10 graphics processor. The display was then evaluated and compared against conventional flight instruments utilizing a six-degree-of-freedom fixed-base simulation. The simulation modeled the Kaman Aerospace Corporation SH-2F "Seasprite" helicopter using a modified Link Aviation Corporation instrument flight trainer and a COMCOR CI-5000 analog computer. Various models of the H-2 helicopter have been used extensively by the U.S. Navy for the all-weather rescue mission. Test subjects used for the evaluation were fleet-experienced Navy helicopter pilots who were familiar with the all-weather rescue mission. They were asked to rate the normal flight instruments and the integrated display systems using the Cooper-Harper Rating System and to offer subjective comments on the systems.

II. SIMULATION

A. MAJOR EQUIPMENT

The major equipment used for the simulation was a Scientific Data Systems model 9300 digital computer, a COMCOR CI-5000 analog computer, an Adage AGT-10 graphics processor and a modified Link Aviation Corporation AC-11B Instrument Flight Trainer. In addition a closed circuit television system was used to transmit the graphics display from the AGT-10 output terminal to a small monitor located in the cockpit of the instrument flight trainer. A schematic drawing of the equipment which was tied together through trunk lines is shown in Figure 1.

B. AIRCRAFT DYNAMICS

The simulated aircraft was the Kaman Aerospace Corporation SH-2F "Seasprite" helicopter which is currently operated by the U.S. Navy in its anti-submarine warfare mission. The flight dynamics of the SH-2F were achieved by solving the aircraft equations of motion in six degrees of freedom using the analog and digital computers. A full development of the equations of motion is contained in Ref. 2 but it will be briefly reviewed here in order to correct some printing errors that exist in the final equations.

Euler's equations of motion of an aircraft subject to aerodynamic and gravity forces are

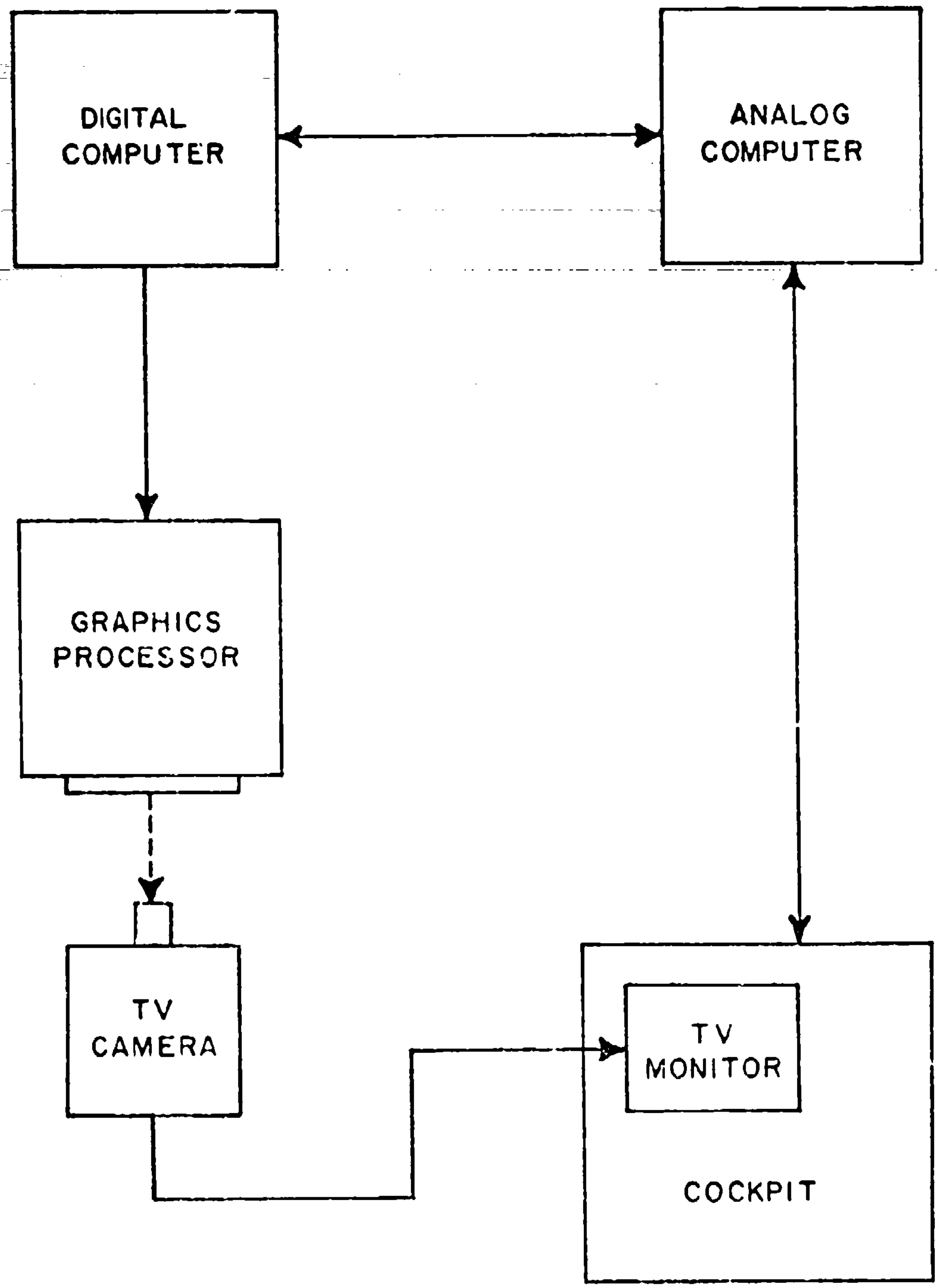


Figure 1. Schematic Diagram of Major Equipment

$$\dot{u} = \frac{X_A}{m} - g \sin\theta + rv - qw$$

$$\dot{v} = \frac{Y_A}{m} + g \sin\phi \cos\theta + pw - ru$$

$$\dot{w} = \frac{Z_A}{m} + g \cos\phi \cos\theta + qu - pv$$

$$\dot{p} = \frac{L_A}{I_{xx}} + \frac{I_{xz}}{I_{xx}} \dot{r}$$

$$\dot{q} = \frac{M_A}{I_{yy}}$$

$$\dot{r} = \frac{N_A}{I_{zz}} + \frac{I_{xz}}{I_{zz}} \dot{p}$$

By applying classical small disturbance theory, assuming no coupling between longitudinal and lateral motions and assuming speed effects on some stability derivatives to be negligible the equations become

$$\dot{u} = X_A(u) + X_A(0) + X_w(0)\, w + X_{B_{1c}}(0)\Delta B_{1c} + X_{\theta_c}(u)\Delta\theta_c - g \sin\theta + rv - qw$$

$$\dot{v} = Y_v(0)\, v + Y_p(0)\, p + Y_r(0)\, r + Y_{A_{1c}}(0)\Delta A_{1c} + Y_{\theta_R}(0)\Delta\theta_R + pw - ru + g \sin\phi \cos\theta$$

$$\dot{w} = Z_A(u) + Z_A(0) + Z_q(0)\, q + Z_w(u)\, w + Z_{B_{1c}}(u)\Delta B_{1c} + Z_{\theta_c}(0)\Delta\theta_c + qu - pv + g \cos\phi \cos\theta$$

$$\dot{p} = \frac{I_{xz}}{I_{xx}} \dot{r} + L_p(0)\, p + L_r(0)\, r + L_v(0)\, v + L_{A_{1c}}(0) \Delta A_{1c} + L_{\theta_R} \Delta \theta_R$$

$$\dot{q} = M_A(u) + M_q(0)\, q + M_w(0)\, w + M_{B_{1c}}(u) \Delta B_{1c} + M_{\theta_c} \Delta \theta_c$$

$$\dot{r} = \frac{I_{xz}}{I_{zz}} \dot{p} + N_p(0)\, p + N_r(u)\, r + N_v(u)\, v + N_{A_{1c}}(0) \Delta A_{1c} + N_{\theta_R} \Delta \theta_R$$

Stability derivatives supplied by the manufacturer were normalized with respect to mass or the appropriate moment of inertia. A corrected listing of the normalized stability derivatives from Ref. 2 is provided in Table I. The derivatives apply to an aircraft with a gross weight of 12,577 pounds at standard sea level atmospheric conditions.

Euler angles were calculated using the following equations [Ref. 7]

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\psi} = q \frac{\sin \phi}{\cos \theta} + r \frac{\cos \phi}{\cos \theta}$$

$$\dot{\phi} = p + \dot{\psi} \sin \theta$$

Inertial velocities were calculated by resolving the body-fixed axes velocities through the Euler angles using the following equations [Ref. 7]

	0 KTS.	30 KTS.	50 KTS.	70 KTS.	91 KTS.	112 KTS.	136 KTS.
$X_A(u)$	0	.2328	-1.501	-3.814	-6.847	-10.62	-15.51
$X_{\theta_c}(u)$	25.06	22.88	29.96	19.94	17.55	17.62	20.45
$Z_A(u)$	0	-9.272	-5.854	.6023	6.554	18.76	34.34
$Z_w(u)$	-.4045	-.5092	-.5843	-.6368	-.6682	-.6875	-.6928
$Z_{B_{1c}}(u)$	4.567	35.35	64.79	97.41	131.1	164.4	197.4
$L_v(u)$	-.0215	-.0261	-.0298	-.0352	-.0409	-.0464	-.0519
$M_A(u)$	0	.1417	.1940	.2244	.2610	.2456	.1835
$M_{B_{1c}}(u)$	-12.17	-12.22	-12.34	-12.54	-12.82	-13.04	-12.15
$N_r(u)$	-.5871	-.7410	-.8881	-1.080	-1.269	-1.447	-1.622
$N_v(u)$	.0172	.0202	.0227	.0272	.0312	.0352	.0399

$X_A(0)$	2.806	$Y_p(0)$	-1.139
$X_q(0)$	.8689	$Y_r(0)$	.9627
$X_w(0)$	.0491	$\bar{Y}_{A_{1c}}(0)$	42.63
$\bar{X}_{B_{1c}}(0)$	40.13	$Y_{\theta_r}(0)$	18.16
$Z_A(0)$	-32.08	$L_p(0)$	-2.425
$Z_q(0)$	.5228	$L_r(0)$	.4082
$\bar{Z}_{\theta_c}(0)$	-208.2	$\bar{L}_{A_{1c}}(0)$	36.51
$M_q(0)$	-.7853	$L_{\theta_r}(0)$	7.075
$M_w(0)$	-.0002	$N_p(0)$	-.0072
$M_{\theta_c}(0)$	.7789	$\bar{N}_{A_{1c}}(0)$	1.877
$Y_v(0)$	-.0338	$N_{\theta_r}(0)$	-11.86

TABLE I. -- STABILITY DERIVATIVES

$$V_x = u \cos\theta \cos\psi + v (\sin\phi \sin\theta \cos\psi - \cos\phi \sin\psi)$$
$$+ w (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)$$

$$V_y = u \cos\theta \sin\psi + v (\sin\phi \sin\theta \sin\psi + \cos\phi \cos\psi)$$
$$+ w (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi)$$

$$V_z = - u \sin\theta + v \sin\phi \cos\theta + w \cos\phi \cos\theta$$

Inertial position which was an important part of the simulation was calculated by numerical integration of the inertial velocities.

C. DIGITAL COMPUTER PROGRAM

The digital computer was the brain of the simulation. It was used for control of the simulation, data storage and access, initialization of the analog and graphics computers, and performance parameter calculations. It provided inputs to the analog computer for solution of the equations of motion and generated the data required for the graphics processor to produce the instrument display. In addition, it was used to calculate inputs for two cockpit instruments, the airspeed indicator and the radar altimeter, which had piece-wise linear scales that precluded a direct connection to an analog output.

A significant problem which developed during initial testing of the integrated display using the analog simulation was stability. Many of the analog inputs are generated in the digital computer by mathematical operations on, or using information derived from, analog outputs. This procedure is cyclic and is termed the dynamic loop. The time required to generate the data for the moving portions of the integrated instrument display, and for the crew directions, caused the total time for the

dynamic loop to exceed the stability limits for the simulation. This resulted in divergent pitch and roll oscillations. A major clean up effort was made in the program to shorten the time required for the FLY, INST and CREW subprograms. This resulted in a significant reduction of the time required to cycle through the dynamic loop which restored stability to the simulation.

A listing of the digital computer program is provided in Appendix A. In addition, a listing of the FORTRAN variables used in the program together with their definitions is provided in Appendix B.

D. ANALOG COMPUTER PROGRAM

The analog computer was used mainly to solve the aircraft equations of motion which were discussed previously. Additionally, it provided a logic interface between cockpit control switches and the digital computer and a direct interface for signals from the digital computer to the cockpit airspeed indicator and radar altimeter. The analog program was essentially the one developed by Hoxie and reported in Ref. 2. Some minor changes were made to the program and are included in Appendix C.

Inputs were provided to the analog computer from the cockpit and from the digital computer through digital to analog converters. Signals from the cockpit were in the form of D.C. voltage signals from simulation control switches and from potentiometers attached to the flight controls. Output signals from the analog computer were used as inputs to cockpit flight instruments and to provide the digital computer with necessary information.

The analog computer patching diagrams and potentiometer settings are contained in Appendix C. Table CI in the Appendix lists the trunk lines running between the analog computer and the cockpit together with the signals carried.

E. GRAPHICS COMPUTER

The graphics computer was used to generate the integrated electronic instrument display which was evaluated. Existing software [Ref. 4] permitted easy interface between the graphics processor and the FORTRAN program used for the digital computer. The instrument display was generated in two steps from data supplied by the digital computer. The static portions (scale marks, numbers, etc.) were generated during the initialization phase of the simulation from data generated by the digital program subroutine DSPLY. Then during the simulation the dynamic portions of the display (pointers, horizon line, etc.) were generated from data provided by the digital program subroutine INST. In addition, simulated directions from the rescue aircrewman were provided to the graphics processor by the digital program subroutine CREW.

F. COCKPIT

The cockpit used was a Link Aviation Corporation AC-11B Instrument Flight Trainer which has undergone extensive modifications. Modifications to allow interfacing with the analog computer used for this evaluation are reported in Refs. 5 and 6. Additional minor modifications to permit the trainer to be used as a helicopter simulator are reported in Ref. 2. One cockpit flight instrument, the direction velocity indicator (DVI), was not available for use in the simulation. This instrument which is similar in appearance to an instrument landing

system (ILS) indicator with the addition of a vertical scale on the left side is used to provide the pilot with velocity information derived from the helicopter's doppler radar. It was easy to simulate this instrument graphically, however, with very little loss of realism so, a graphical DVI was used in conjunction with the normal cockpit instrument display. Its location on the TV monitor placed it outside the optimal scan pattern when using the normal flight instruments, however, this was thought to be a minor inconvenience which would not appreciably affect the results of the evaluation.

Simulation control switches utilized by Hoxie [Ref. 2] were also used for this evaluation with the exception that the NORMAL-AUTOMATIC switch which was used to control the type of instrument display generated by the graphics processor was changed to NORMAL-INTEGRATED. Its function was to select the type of instrument display desired. The NORMAL position selected the cockpit flight instruments and the INTEGRATED position selected the integrated display in addition to the cockpit flight instruments.

Signals carried by the trunk lines between the cockpit and the analog computer are shown in Table CI.

III. INTEGRATED ELECTRONIC INSTRUMENT DISPLAY

The integrated electronic display which was evaluated was developed with two basic ideas in mind. First, was to design a display which would be compact. This would reduce pilot scan time and hopefully increase the accuracy of aircraft control which would result in a more precise hover. Second, was to keep the display format as close as possible to that of conventional flight instruments. This would reduce or eliminate any pilot learning or adjustment time and permit easy transition from conventional flight instruments to the integrated display.

The display which was developed is shown in Figure 2 for two flight conditions. The top photograph shows the helicopter in a level cruise condition at 500 feet and 70 knots. The bottom photograph depicts the helicopter in a stable hover at an altitude of 40 feet. Figure 2 shows the display at approximately one-half actual size.

The reasoning behind the display design was simple but hopefully straightforward. Radar height information was depicted as a vertical scale with a sliding pointer. This was to provide a clearer analog correspondence to the actual physical situation -- height above the ground -- than that provided by conventional circular instruments. It is located on the left side of the display to provide direct correspondence with the flight control which is primarily used to control altitude -- the collective pitch lever. The airspeed scale, likewise, is located on the right side of the display to correspond to the right-hand flight control -- cyclic pitch.

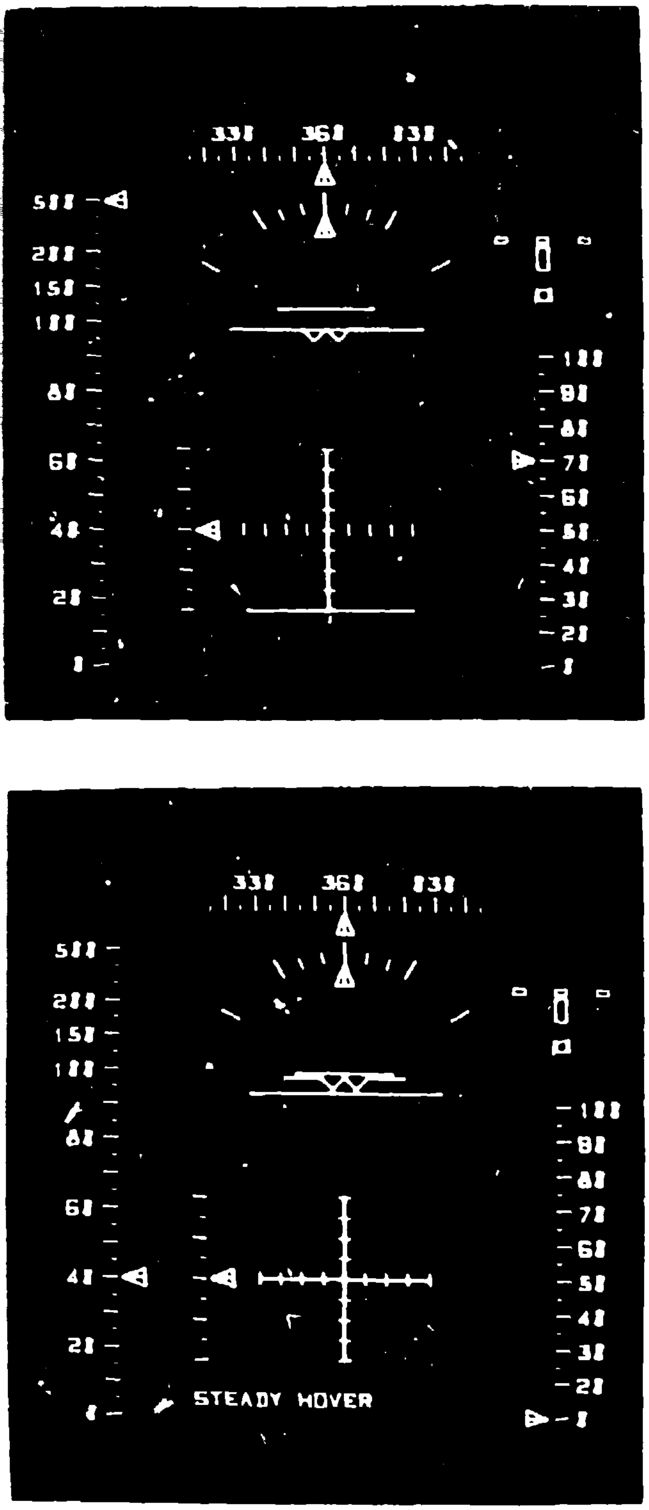


Figure 2. Integrated Electronic Instrument Display

Velocity information derived from the helicopter's doppler radar was displayed in a format almost identical to that of the conventional DVI instrument. Normal cockpit flight instrument arrangements have the DVI located above the attitude gyro indicator; however, in the integrated display it was located below the attitude gyro next to the radar altimeter scale. The reason for this can be understood by examining Figure 2. Note that in the situation shown in the bottom photograph, which is the desired condition for a precision hover, the radar altimeter pointer, the doppler vertical speed pointer and the doppler fore-aft ground speed line are in a horizontal line across the bottom of the display. Any deviation from the desired hover condition should immediately be noticed by the pilot as a deviation of this "line". This should decrease reaction time in correcting for deviation of these parameters and, therefore, result in a more precise hover.

Pitch and roll attitude information was portrayed in a conventional format and since a central position was desired it was located at the top center of the display. Turn rate and slip information which is almost useless in a precision hover was depicted in nearly conventional format in the upper right area of the display. The turn needle slides horizontally under the standard rate turn scale marks rather than rotating about its lower end as in a conventional turn indicator. Turn and slip information is, however, important for the instrument approach flight condition which is discussed later.

Heading information was displayed as a horizontal tape sliding past an index pointer which indicated actual heading. The tape moved as one would expect (i.e. right to left for a right turn) instead of backwards which is the characteristic of the old-style horizontal directional

gyros found in older aircraft. Most pilots prefer the circular vertical card display for heading, however, limitations in the graphics processor precluded displaying a circular compass rose which could rotate. This situation could probably be solved if a special purpose integrated electronic display were to be developed. Since heading information is of minor importance during the precision hover task due to good heading hold features of helicopter automatic stabilization equipment this limitation was considered to be a very minor drawback to the evaluation.

During development of the display the digital computer program which generates the graphics data for the display was intentionally kept as general as possible. This permitted such parameters as location, size, and spacing of the individual instruments to be changed with a minimum of program changes. Many combinations of the above parameters were tried before arriving at the final display shown in Figure 2. The basic display is the author's idea of how it should look; however, suggestions concerning the display were solicited from experienced Navy helicopter pilots during its development. These suggestions resulted in some rearrangement of the individual instruments which, hopefully, led to a practical, well-designed display.

IV. EVALUATION

A. TASK

The final approach and precision hover phases of a night over-water rescue mission were simulated for the evaluation. The aircraft was situated two nautical miles from the target at an altitude of 500 feet, heading toward the target at a speed of 70 knots. This permitted time for the test subject to become comfortable flying the simulator and to descend to 150 feet before reaching the one nautical mile "gate" position. At the gate position the pilot commenced a descent and a deceleration to arrive over the survivor in a hover at an altitude of 40 feet. Since a rescue aircrewman was not available, appropriate directions to the pilot were supplied via the integrated display in the form of standard movement commands [Ref. 3] to position the aircraft over the survivor. The pilot was required to maintain a hover position within an area of ± 30 feet of the survivor for two continuous minutes to effect the pick-up. An additional 30 seconds were then required to hoist the survivor aboard. Timing was not started until the aircraft was initially maneuvered to within ± 9 feet of the survivor. If the helicopter drifted outside the ± 30 feet area timing was stopped and then restarted at zero whenever the aircraft was again manouvered to within ± 9 feet of the survivor. After retrieving the survivor the pilot departed straight ahead climbing to 500 feet. This completed the task.

B. PILOTS

Five fleet-experienced Navy helicopter pilots were used for the evaluation. All of them were familiar with the over-water rescue

mission and all of them had flown either or both of the U.S. Navy model H-2 or H-3 helicopters operationally. Both of these helicopters are used for over-water rescue missions and both have similar cockpit flight instrument displays. Pilot experience is listed in Appendix D together with the ratings they assigned to the two instrument displays.

C. FAMILIARIZATION

Each evaluation pilot was given a short briefing on the simulation and the equipment. He was told basically how it worked and how it could be controlled from the cockpit. He was advised of the peculiarities of the trainer and how it would differ from a real helicopter. He was told what the evaluation task would be and what would be expected of him. The Cooper-Harper rating scale (Figure 3) was explained and he was asked to keep this scale in mind while flying the simulation.

Each pilot was then given the opportunity to fly the helicopter until he was comfortable with its handling qualities and familiar with the instrumentation. During this time he made several practice approaches to a hover using both instrument systems.

D. EVALUATION FLIGHTS

After sufficient familiarization the pilots each made two flights for the record during which they completed the evaluation task. During one flight conventional cockpit flight instruments were used and during the other the integrated display was used. During the flights, "performance indicators" in the form of root mean square values of inertial velocities and inertial position were computed by numerical integration using the digital computer. Computation of these values was started whenever the helicopter was first within ± 9 feet of the survivor and

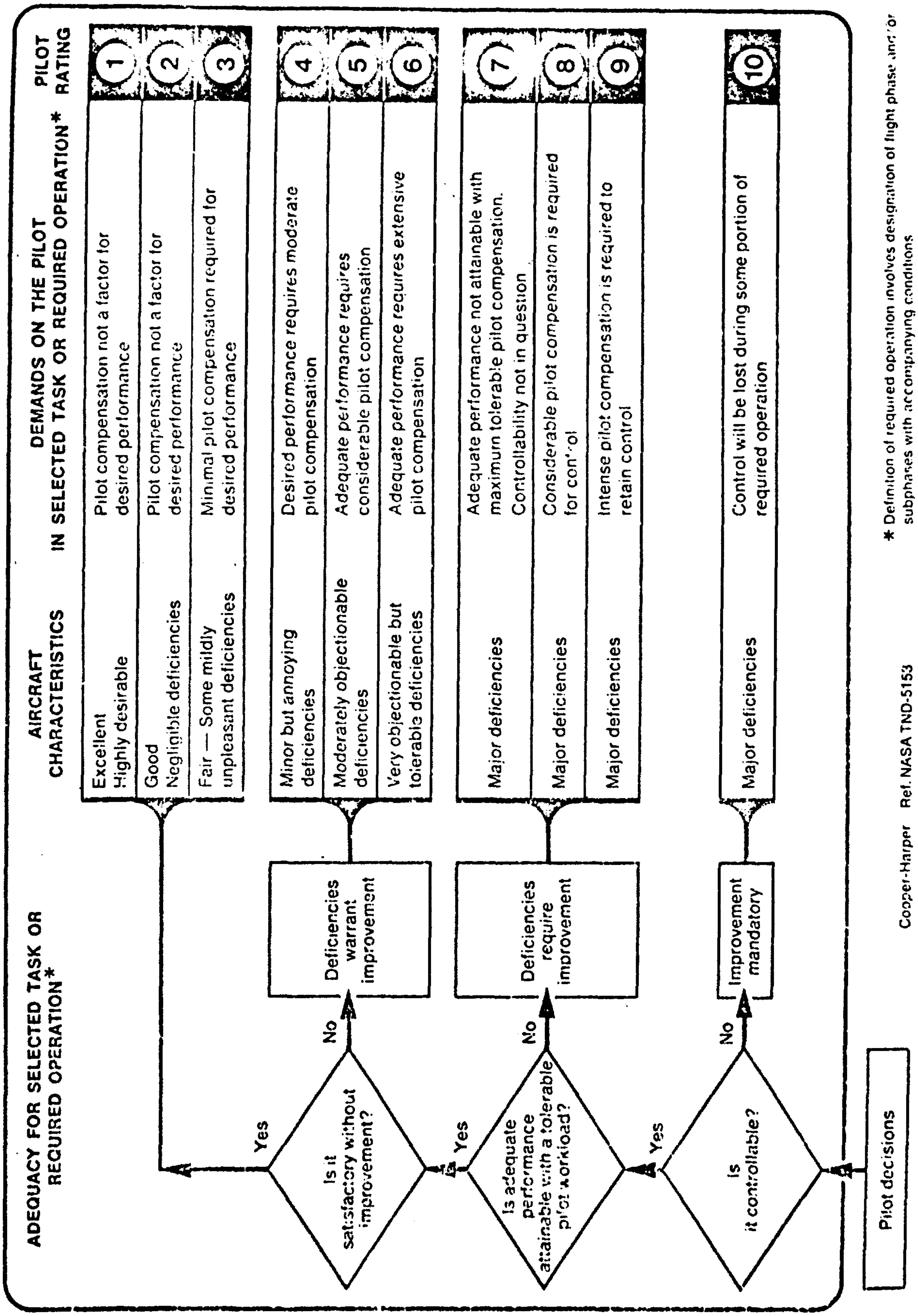


Figure 3. Cooper-Harper Rating Scale

continued until the rescue was effected. Computation did not restart if the helicopter drifted outside the pick-up area.

At the end of the flights each pilot was asked to rate the conventional cockpit flight instrument display using the Cooper-Harper rating scale. Then using this rating as a base line he was asked to assign a rating to the integrated display. He was also asked to give any subjective comments concerning the integrated display that he cared to make. The ratings are listed in Appendix D.

V. CONCLUSIONS

The primary conclusion of the evaluation was that the aircraft dynamics portion of the simulation was unsatisfactory and did not adequately represent a real helicopter in the hover flight condition. Four of the five evaluation pilots were not able to complete the evaluation task within a reasonable time period. The simulated aircraft was particularly susceptible to pilot induced oscillations in pitch and roll while in a hover. These oscillations tended to be slightly divergent which resulted in the pilot being unable to maintain the required hover position for the required time. In a few instances complete loss of control resulted. Maintaining control required intense pilot concentration on pitch and roll attitude which often resulted in large altitude excursions and the frequent display of the "PULL UP -- YOU ARE LOW" crew command. Although Hoxie /Ref. 2/ reported the aircraft simulation to be satisfactory, examination of the pilot task used in that evaluation showed that the pilot was required to maintain a position within ± 40 feet of the target and to maintain it for only 45 seconds. Timing was started as soon as the aircraft was within the prescribed area. The target area dimensions used by Hoxie were more than 30% larger than those used in this evaluation. In addition, the 45 second time period required to complete the task is unrealistically low. It is felt that even the two minutes used in this evaluation is very low compared with that which is encountered in an actual night over-water rescue. The aircraft simulation is, therefore, unsatisfactory for evaluations of this type if realistic parameters for hover time and position are specified.

All of the evaluation pilots preferred the integrated display over the normal cockpit instruments, but some were more enthusiastic toward it than others. Areas receiving the most comments were the radar altitude scale and the relative locations of the radar altitude scale and the doppler vertical speed scale. The evaluation pilots preferred the vertical style scale used for radar altitude over conventional circular instruments. They felt it was a more meaningful representation of altitude. Four of the five pilots commented on the relative location of the radar altimeter pointer and the doppler vertical speed pointer in the hover flight condition. They liked the fact that in a hover the pointers were close and lined up so that deviations could easily be noted. Several pilots commented that these scales were particularly easy to interpret.

The area of the display which received the most negative comment was the attitude gyro. Three pilots thought that it should be larger so that small attitude changes could more easily be made and that small deviations could more easily be noticed. It should be noted that helicopter pilots are particularly aware and concerned with attitude changes especially in a hover. In fact most helicopters are equipped with attitude gyro indicators which are much larger than those used in most fixed wing aircraft. One pilot commented on the lack of three-dimensional depth for the integrated display attitude indicator. This was probably due to his familiarity with the conventional attitude indicator which displays attitude as the rotation of a sphere about two perpendicular horizontal axes. Two pilots suggested the incorporation of additional pitch scale lines on the indicator to indicate divisions of five degrees.

As was mentioned earlier only one of the pilots, Pilot B, was able to complete the task in a reasonable time period of practice. Table II shows the performance indicators which were calculated for Pilot B.

	V_x(fps)	V_y(fps)	V_z(fps)	X_E(ft)	Y_E(ft)	Z_E(ft)
Normal Instruments	1.38	.43	7.03	10.4	8.4	76.4
Integrated Display	1.91	1.48	3.23	16.4	20.4	47.4
Ideal	~0.	~0.	~0.	~0.	~0.	40.0

TABLE II. PILOT B PERFORMANCE INDICATORS

Note that when Pilot B was flying using the normal instruments his attention was evidently concentrated on maintaining minimum horizontal velocities. This resulted in small horizontal position deviations but large altitude variations. When flying while using the integrated display his altitude control was significantly better but horizontal control was less. Closer control of altitude deviations is also evident in strip chart recordings which were made of Pilot B's collective stick control inputs. These recordings which have identical scales are shown in Figure 4. Note that the frequency of significant control inputs is much higher for the integrated display than for the normal display. Pilot B commented that when flying while using the integrated display he was forced to pay more attention to altitude control because his peripheral vision picked up the movement of the radar altitude and doppler vertical speed pointers. He mentioned that his scan tended to break down when using either display and that he would fixate on the DVI because of the intense control required to accomplish the task.

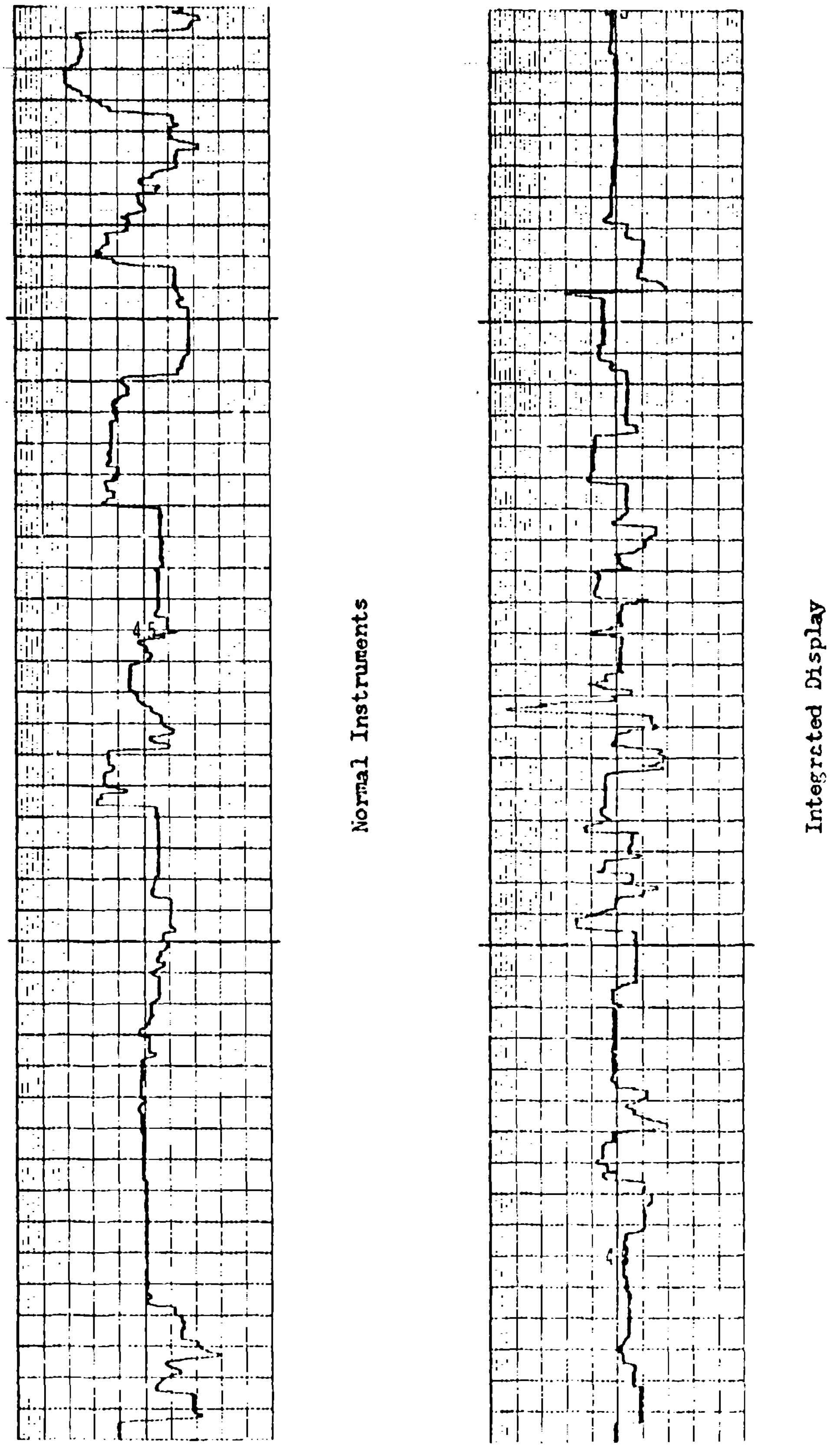


Figure 4. Pilot B Collective Stick Control Inputs

It was concluded that an integrated electronic instrument display such as the one evaluated would probably be an asset to safer, more precise helicopter hover operations. However, due to the difficulties experienced with the aircraft dynamics used in the simulation, additional study should be undertaken with this or similar integrated displays before any accurate conclusions are drawn.

The display which was developed for this evaluation could easily be modified for other uses in a helicopter. The DVI could easily be converted for use as an instrument landing system (ILS) indicator or as a TACAN course deviation indicator (CDI). The radar altitude scale could also be changed and used to display barometric altitude when the aircraft is flying at higher altitudes.

It was suggested by one pilot that since a display of this type would probably be generated by a special purpose micro-computer its potential was almost limitless. He suggested displaying a target indicator, which would be set upon first passing over the target, whose position relative to the center of the DVI would indicate the target's actual position relative to the aircraft. Target position would be generated from the initial position and the integrated doppler velocities. This would virtually eliminate the need for continuous crew-to-pilot communications now necessary for an efficient rescue operation.

Most military helicopter pilots are aware of the advances which have been made in the areas of integrated electronic and "heads-up" instrument displays for fixed wing aircraft and they are concerned that little is being done to help alleviate their work load using similar technology. An area ripe for application of this technology is the precision hover conducted while flying solely by reference to instruments.

APPENDIX A

DIGITAL COMPUTER PROGRAM

This appendix contains a listing of the digital computer program which is written in the Scientific Data Systems FORTRAN IV language.

```
*****************************************************************
*                                                               *
*  EVALUATION OF AN INTEGRATED ELECTRONIC INSTRUMENT DISPLAY FOR *
*                                                               *
*    HELICOPTER HOVER OPERATIONS USING A SIX-DEGREE OF FREEDOM   *
*                                                               *
*         FIXED-BASE SIMULATION OF THE SH-2F HELICOPTER          *
*                                                               *
*****************************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      DIMENSION DRVNAM(20)
      COMMON DRVTAB(10,7),CRWDIR(7,17),B
      COMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      COMMON /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDOT,
     1               PHIDOT,PSIDOT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      COMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1              XCHDG,RF,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2              YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3              XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DATA NULL/-1/,ALTNUM/18*-1/,AIRNUM/20*-1/,CMPNUM/90*-1/,
     1     IBLANK/2*0/,HGT/.07/,BASE/.06/,SCALE/.20/,ASCALE/.40/,
     2     DCMPS/.05/,YOHDG/.87/,INDYM(1)/3777B/,INDYM(83)/0/
      NAMELIST IDEV,U,V,W,PHI,THETA,PSI,XE,YE,ZE,UKTS,B
*
*  READ STABILITY DERIVATIVES - PRINT TABULAR LISTING
*
      READ(5,100) DRVNAM
  100 FORMAT(20A4)
      READ(5,101)((DRVTAB(I,J),J=1,7),I=1,10)
  101 FORMAT(7F10.4)
      WRITE(6,102)
  102 FORMAT('1',/////,40X,'STABILITY DERIVATIVES'//)
      WRITE(6,103)
```

```
103 FØRMAT('0',20X,'0 KTS',5X,'30 KTS',5X,'50 KTS',5X,'70 KTS',5X,
   1        '91 KTS',4X,'112 KTS',4X,'136 KTS',/)
    WRITE(6,104)(DRVNAM(2*I-1),DRVNAM(2*I),(DRVTAB(I,J),J=1,7),I=1,10)
104 FØRMAT('0',8X,2A4,F9.4,6F11.4)
*
*  READ CREW DIRECTIØNS - PRINT LISTING
*
    READ(5,105)((CRWDIR(I,J),I=1,6),J=1,17)
105 FØRMAT(6A4)
    WRITE(6,106)
106 FØRMAT('1',/////,41X,'CREW DIRECTIØNS'/)
    WRITE(6,107)((CRWDIR(I,J),I=1,6),J=1,17)
107 FØRMAT('0',17X,18A4)
*
*  READ RADAR ALTIMETER SCALE NUMBERS - ECHØ CHECK
*
    READ(5,108)(ALTNUM(J),J=1,17,2)
108 FØRMAT(20A4)
    WRITE(6,109)
109 FØRMAT(//'0',17X,'ALTITUDE SCALE NUMBERS')
    WRITE(6,110)(ALTNUM(J),J=1,17,2)
110 FØRMAT('0',14X,20A4//)
*
*  READ AIRSPEED SCALE NUMBERS - ECHØ CHECK
*
    READ(5,108)(AIRNUM(J),J=1,19,2)
    WRITE(6,111)
111 FØRMAT('0',17X,'AIRSPEED SCALE NUMBERS')
    WRITE(6,112)(AIRNUM(J),J=1,19,2)
112 FØRMAT('0',17X,20A4//)
*
*  READ COMPASS NUMBERS - ECHØ CHECK
*
    READ(5,113)(CMPNUM(J),J=1,90)
113 FØRMAT(48A1)
```

```
      WRITE(6,114)
  114 FØRMAT('0',17X,'CØMPASS NUMBERS')
      WRITE(6,115)(CMPNUM(J),J=1,90)
  115 FØRMAT('0',16X,48A1,/,' ',16X,48A1)
*
*  SET PØTENTIØMETERS
*
      CALL SETPØT (4HP000,0890, 4HP001,4724, 4HP002,1320, 4HP003,2006,
     1             4HP004,0869, 4HP005,1962, 4HP006,0800, 4HP007,0000,
     2             4HP010,0200, 4HP011,1224, 4HP012,2500, 4HP013,1250,
     3             4HP014,0100, 4HP015,0833, 4HP016,0250, 4HP017,7075,
     4             4HP020,7853, 4HP021,0389, 4HP022,1066, 4HP023,0241,
     5             4HP024,2000, 4HP025,5000, 4HP026,0285, 4HP027,0845,
     6             4HP030,4000, 4HP031,1316, 4HP032,9127, 4HP033,1021,
     7             4HP034,6062, 4HP035,2598, 4HP036,4000, 4HP037,2500,
     8             4HP040,1039, 4HP041,0018, 4HP042,4692, 4HP043,1186,
     9             4HP044,4000, 4HP045,1250, 4HP046,6250, 4HP047,2500,
     A             4HP050,3200, 4HP051,1041, 4HP052,2000, 4HP053,1046,
     B             4HP054,1052, 4HP055,1258, 4HP056,5000, 4HP057,1128)
*
*  SELECT GRAPHICS CØMPUTER
*
      ØUTPUT(101) 'SELECT GRAPHICS CØMPUTER - TYPE: IDEV=  *'
      INPUT(101)
*
*  SET INITIAL FLIGHT CØNDITIØNS
*
  200 CALL RESET(1000)
      UKTS=70.
      U=UKTS*1.687
      V=0.
      W=5.26
      P=0.
      Q=0.
      R=0.
```

```
      PHI=0.
      THETA=2.55/57.3
      PSI=0.
      PHIDOT=0.
      THEDOT=0.
      PSIDOT=0.
      XE=-2.*2000.
      YE=0.
      ZE=500.
      B=.3
*
*  INPUT ALTERNATE INITIAL CONDITIONS IF DESIRED
*
      IF(SENSE SWITCH 3) 202,205
  202 OUTPUT(101) 'INPUT ALTERNATE INITIAL CONDITIONS'
      INPUT(101)
*
*  INITIALIZE GRAPHICS COMPUTER
*
  205 NTD=20
      NGD=20
      CALL DTINIT(IDEV,ITD,NTD,IER)
      IF(IER.NE.0) OUTPUT(6) 'ERROR - DTINIT',IER
      CALL DGINIT(IDEV,IGD,NGD,IER)
      IF(IER.NE.0) OUTPUT(6) 'ERROR - DGINIT',IER
*
*  GENERATE STATIC PORTION OF INSTRUMENT DISPLAY
*
      CALL DSPLY
*
*  COMMENCE DYNAMIC SIMULATION
*
      CALL FLY
*
*  COMPUTE RMS INERTIAL VELOCITIES AND RMS POSITIONS
```

```
*
      VXRMS=SQRT(VXRMS/RMSTIM)
      VYRMS=SQRT(VYRMS/RMSTIM)
      VZRMS=SQRT(VZRMS/RMSTIM)
      XERMS=SQRT(XERMS/RMSTIM)
      YERMS=SQRT(YERMS/RMSTIM)
      ZERMS=SQRT(ZERMS/RMSTIM)
      ØUTPUT(6) ' ',VXRMS,VYRMS,VZRMS,XERMS,YERMS,ZERMS,RMSTIM,FLTIM
*
*  CHECK FØR QUIT ØR RERUN
*
  210 IF(TEST(5).LT.0) GØ TØ 200
      IF(TEST(4).LT.0) GØ TØ 220
      GØ TØ 210
  220 CALL PØTSET
      STØP
      END
```

```
      SUBROUTINE DSPLY
***********************************************************
*                                                         *
*  GENERATES STATIC PORTION OF INSTRUMENT DISPLAY         *
*                                                         *
***********************************************************
      INTEGER CRADIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      COMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      COMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DIMENSION IVSI(20),IDVI(38),IALT(59),IVGI(20),IMAP(9),ITURN(17),
     1          ISLIP(6),ISPD(38)
      NBLK=1
*
*  VERTICAL SPEED INDICATOR - SCALE DIVISIONS
*
      XOVSI=-2.50*SCALE
      YOVSI=-1.95*SCALE
      DS=.10*SCALE
      DL=.22*SCALE
      DVSI=.35*SCALE
      Y=YOVSI-4.*DVSI
      IVSI(1)=IHEAD(0,10)
      IVSI(2)=IPACK(XOVSI,Y,0)
      DO 110 J=3,18,4
        X=XOVSI+DL
        IVSI(J)=IPACK(X,Y,1)
        X=X-DS
        Y=Y+DVSI
        IVSI(J+1)=IPACK(X,Y,0)
        X=X+DS
        IVSI(J+2)=IPACK(X,Y,1)
```

```
      X=X-DL
      Y=Y+DVSI
      IVSI(J+3)=IPACK(X,Y,0)
110   CONTINUE
     X=X+DL
     IVSI(19)=IPACK(X,Y,1)
     IVSI(20)=0
     CALL GRAPH9(IDEV,IVSI,20,NBLK,IER)
     NBLK=NBLK+1
     IF(IER.NE.0) OUTPUT(6) 'ERROR--IVSI',IER,*
     XPVSI=X+.10*SCALE
*
*  DIRECTION VELOCITY INDICATOR - SCALE DIVISIONS
*
     XODVI=0.
     YODVI=YOVSI
     DL=0.20*SCALE
     DDVI=0.35*SCALE
     X=XODVI-4.*DDVI
     Y=YODVI-DL/2.
     IDVI(1)=IHEAD(0,10)
     IDVI(2)=IPACK(X,Y,0)
     DO 115 J=3,19.2
       Y=Y+DL
       IDVI(J)=IPACK(X,Y,1)
       X=X+DDVI
       Y=Y-DL
       IDVI(J+1)=IPACK(X,Y,0)
115   CONTINUE
     X=XODVI-DL/2.
     Y=YODVI-4.*DDVI
     IDVI(20)=IPACK(X,Y,0)
     DO 120 J=21,37,2
       X=X+DL
       IDVI(J)=IPACK(X,Y,1)
```

```
      X=X-DL
      Y=Y+DDVI
      IDVI(J+1)=IPACK(X,Y,0)
 120  CONTINUE
     IDVI(38)=0
     CALL GRAPH8(IDEV,IDVI,38,NBLK,IER)
     NBLK=NBLK+1
     IF(IER.NE.0) OUTPUT(6) 'ERROR--IDVI',IER,*
     XLEFT= XODVI-4.*DDVI
     XRIGHT=XODVI+4.*DDVI
     YTOP=YODVI+4.0*DDVI
     YBOT=YODVI-4.0*DDVI
*
*  CHECK FOR DESIRED DISPLAY
*
     IF(TEST(6).GE.0) RETURN
*
*  RADAR ALTIMETER - SCALE NUMBERS
*
     LNO=36
     LN=LNO
     ICOPS=6
     DO 108 J=1,11,2
       CALL TEXT8(IDEV,ALTNUM(J),2,LN,ICOPS,2,3,IER)
       IF(IER.NE.0) OUTPUT(6),'ERROR--ALTIMETER SCALE NUMBERS',J,IER,*
       LN=LN-4
 108   CONTINUE
     LN=LN+2
     CALL TEXT8(IDEV,ALTNUM(13),2,LN,ICOPS,2,3,IER)
     IF(IER.NE.0) OUTPUT(6),'ERROR--ALTIMETER SCALE NUMBER 7',IER,*
     LN=LN-2
     CALL TEXT8(IDEV,ALTNUM(15),2,LN,ICOPS,2,3,IER)
     IF(IER.NE.0) OUTPUT(6),'ERROR--ALTIMETER SCALE NUMBER 8',IER,*
     LN=LN-3
     CALL TEXT8(IDEV,ALTNUM(17),2,LN,ICOPS,2,3,IER)
```

```
      IF(IER.NE.0) OUTPUT(6) 'ERROR--ALTIMETER SCALE NUMBER 9',IER,*
*
*  RADAR ALTIMETER - SCALE
*
      XOALT=.125*(ICOPS-40.)*SCALE
      YOALT=.3*(21.5-LNO)*SCALE
      DS=.10*SCALE
      DL=.25*SCALE
      DALT=.30*SCALE
      Y=YOALT
      IALT(1)=IHEAD(0,10)
      IALT(2)=IPACK(XOALT,YOALT,0)
      DO 102 J=3,51,4
        X=XOALT+DL
        IALT(J)=IPACK(X,Y,1)
        X=X-DS
        Y=Y+DALT
        IALT(J+1)=IPACK(X,Y,0)
        X=X+DS
        IALT(J+2)=IPACK(X,Y,1)
        X=X-DL
        Y=Y+DALT
        IALT(J+3)=IPACK(X,Y,0)
  102   CONTINUE
      X=X+DL-DS
      IALT(55)=IPACK(X,Y,0)
      X=X+DS
      IALT(56)=IPACK(X,Y,1)
      X=X-DL
      Y=Y+DALT
      IALT(57)=IPACK(X,Y,0)
      X=X+DL
      IALT(58)=IPACK(X,Y,1)
      IALT(59)=0
      CALL GRAPH0(IDEV,IALT,59,NBLK,IER)
```

```
      NBLK=NBLK+1
      IF(IER.NE.0) BUTPUT(6) 'ERRBR--IALT',IER,*
      XPALT=X+.10*SCALE
*
*  VERTICAL GYRB INDICATBR - ANGLE BF BANK SCALE
*
      XCEN=0.
      YCEN= 1.5*SCALE
      R=2.0*SCALE
      IVGI(1)=IHEAD(0,10)
      J=2
      REPEAT 122, FBR RBLL=-60.,-30.,-20.,-10.,0.,10.,20.,30.,60.
        RL=2.35*SCALE
        ARBLL=ABS(RBLL)
        IF(ARBLL.EQ.10..BR.ARBLL.EQ.20.) RL=2.20*SCALE
        Y=YCEN+R*CBS(RBLL/57.3)
        X=XCEN+R*SIN(RBLL/57.3)
        IVGI(J)=IPACK(X,Y,0)
        X=XCEN+RL*SIN(RBLL/57.3)
        Y=YCEN+RL*CBS(RBLL/57.3)
        IVGI(J+1)=IPACK(X,Y,1)
        J=J+2
  122   CBNTINUE
      IVGI(20)=0
      CALL GRAPHR(IDEV,IVGI,20,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) BUTPUT(6) 'ERRBR--IVGI',IER,*
      RP=R-.05*SCALE
      RB=RP-HGT
      EPS=ATAN(BASE/(2.*RB))
*
*  VERTICAL GYRB INDICATBR - MINIATURE AIRPLANE
*
      DL=1.0*SCALE
      DS=0.2*SCALE
```

```
      IMAP(1)=IHEAD(0,10)
      X=XCEN-DL
      IMAP(2)=IPACK(X,YCEN,0)
      X=XCEN-2.*DS
      IMAP(3)=IPACK(X,YCEN,1)
      X=XCEN-DS
      Y=YCEN-DS
      IMAP(4)=IPACK(X,Y,1)
      IMAP(5)=IPACK(XCEN,YCEN,1)
      IMAP(6)=IPACK(-X,Y,1)
      X=XCEN+2.*DS
      IMAP(7)=IPACK(X,YCEN,1)
      X=XCEN+DL
      IMAP(8)=IPACK(X,YCEN,1)
      IMAP(9)=0
      CALL GRAPHP(IDEV,IMAP,9,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--IMAP',IER,*
*
*  SLIP INDICATOR - SCALE
*
      XOSLIP= 3.60*SCALE
      YOSLIP= 2.10*SCALE
      DS= .25*SCALE
      X=XOSLIP-DS/2.
      Y=YOSLIP+DS/2.
      ISLIP(1)=IHEAD(0,10)
      ISLIP(2)=IPACK(X,Y,0)
      Y=Y-DS
      ISLIP(3)=IPACK(X,Y,1)
      X=X+DS
      ISLIP(4)=IPACK(X,Y,0)
      Y=Y+DS
      ISLIP(5)=IPACK(X,Y,1)
      ISLIP(6)=0
```

```
      CALL GRAPH9(IDEV,ISLIP,6,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--ISLIP',IER,*
*
*  RATE OF TURN - SCALE
*
      XOTURN=XOSLIP
      YOTURN=YOSLIP+0.90*SCALE
      DS=.10*SCALE
      DL=.20*SCALE
      DTURN=.70*SCALE
      ITURN(1)=IHEAD(0,10)
      X=XOTURN-DTURN+DL/2.
      Y=YOTURN
      ITURN(2)=IPACK(X,Y,0)
      DO 135 J=3,13,5
        X=X-DL
        ITURN(J)=IPACK(X,Y,1)
        Y=Y+DS
        ITURN(J+1)=IPACK(X,Y,1)
        X=X+DL
        ITURN(J+2)=IPACK(X,Y,1)
        Y=Y-DS
        ITURN(J+3)=IPACK(X,Y,1)
        X=X+DTURN
        ITURN(J+4)=IPACK(X,Y,0)
135     CONTINUE
      ITURN(17)=0
      CALL GRAPH9(IDEV,ITURN,17,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--ITURN',IER,*
*
*  AIRSPEED - SCALE NUMBERS
*
      LNO=36
```

```
      LN=LNO
      ICØPS=80
      DØ 130 J=1,20,2
        CALL TEXTB(IDEV,AIRNUM(J),2,LN,ICØPS,2,3,IER)
        IF(IER.NE.0) ØUTPUT(6),'ERRØR--AIRPSEED SCALE NUMBERS',J,IER,*
        LN=LN-2
 130    CØNTINUE
*
*  AIRSPEED - SCALE DIVISIØNS
*
      XOSPD=.125*(ICØPS-52.)*SCALE
      YOSPD=.3*(21.5-LNO)*SCALE
      DS=.10*SCALE
      DL=.25*SCALE
      DSPD=.30*SCALE
      ISPD(1)=IHEAD(0,10)
      ISPD(2)=IPACK(XOSPD,YOSPD,0)
      X=XOSPD+DL
      ISPD(3)=IPACK(X,YOSPD,1)
      X=X-DL
      Y=YOSPD+2.*DSPD
      ISPD(4)=IPACK(X,Y,0)
      DØ 125 J=5,33,4
        X=X+DL
        ISPD(J)=IPACK(X,Y,1)
        X=X-DL
        Y=Y+DSPD
        ISPD(J+1)=IPACK(X,Y,0)
        X=X+DS
        ISPD(J+2)=IPACK(X,Y,1)
        X=X-DS
        Y=Y+DSPD
        ISPD(J+3)=IPACK(X,Y,0)
 125    CØNTINUE
      X=X+DL
```

```
ISPD(37)=IPACK(X,Y,1)
ISPD(38)=0
CALL GRAPHB(IDEV,ISPD,38,NBLK,IER)
NBLK=NBLK+1
IF(IER.NE.0) OUTPUT(6) 'ERROR--ISPD',IER,*
XPSPD=X-DL-.10*SCALE
RETURN
END
```

```
      SUBROUTINE FLY
*****************************************************
*                                                   *
*  DYNAMIC SIMULATION                               *
*                                                   *
*****************************************************
      REAL MAU,MB1CU,LVU,NRU,NVU
      INTEGER CRNDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      COMMON DRVTAB(10,7),CRNDIR(7,17),3
      COMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      COMMON /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDOT,
     1               PHIDOT,PSIDOT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      COMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DIMENSION DRV(10)
      EQUIVALENCE (XAU,DRV(1)),(ZAU,DRV(2)),(MAU,DRV(3)),(ZWU,DRV(4)),
     1            (MB1CU,DRV(5)),(LVU,DRV(6)),(NRU,DRV(7)),(NVU,DRV(8)),
     2            (XTHCU,DRV(9)),(ZB1CU,DRV(10))
*
*  SET INITIAL CONDITIONS - SAVE INITIAL VALUES
*
      CALL WRITECLOCK(0)
      IFLAG=0
      JFLAG=0
      KFLAG=0
      FLTIM=0.
      RMSTIM=0.
      VXRMS=0.
      VYRMS=0.
      VZRMS=0.
      XERMS=0.
```

```
      YERMS=0.
      ZERMS=0.
      TØLD=0.
      THETIC=THETA
      DA1C=0.
      DTHR=0.
      KSAV=0
*
  100 IF(IFLAG.EQ.1) GØ TØ 110
      GØ TØ 115
  110 CALL CØMPUTE
      CALL STARTCLØCK
*
*   PRINT AIRCRAFT STATE VARIABLES IF DESIRED
*
  115 IF(SENSE SWITCH 1) 120,130
  120 WRITE(6,125) UKTS,V,W,P,Q,R,PHI,THETA,PSI,XE,YE,ZE,FLTIM
  125 FØRMAT('0',F5.1,12F9.2)
*
*   CALCULATE VALUES ØF AIRSPEED DEPENDENT STABILITY DERIVATIVES
*
  130 DØ 137 I=1,10
         IF(UKTS.LE.0.) DRV(I)=DRVTAB(I,1) ; GØ TØ 137
         J=1
         U1=0.
         REPEAT 135, FØR U2= 30.,50.,70.,91.,112.,136.
         IF(UKTS.GE.U2) GØ TØ 132
         DRV(I)=DRVTAB(I,J)+(DRVTAB(I,J+1)-DRVTAB(I,J))*(UKTS-U1)/(U2-U1)
         GØ TØ 137
  132    J=J+1
         U1=U2
  135    CØNTINUE
         DRV(I)=DRVTAB(I,7)
  137    CØNTINUE
      IF(SENSE SWITCH 2) 500,502
```

```
  500 WRITE(6,501) UKTS,XAU,ZAU,LVU
  501 FORMAT('0',4F12.4)
  502 CONTINUE
*
*  SCALE STABILITY DERIVATIVES
*
      XAU=XAU*.05
      ZAU=ZAU*.01
*     MAU IS OK
      ZWU=ZWU*.5
      MB1CU=-MB1CU*.01
      LVU=LVU*19.23
      NRU=NRU*.25
      NVU=NVU*25.
      XTHCU=XTHCU*.025
      ZB1CU=ZB1CU*.001
*
*  COMPUTE APPROPRIATE TRIG FUNCTIONS TO SAVE COMPUTER TIME
*
      SINPHI=SIN(PHI)
      COSPHI=COS(PHI)
      SINTHE=SIN(THETA)
      COSTHE=COS(THETA)
      SINPSI=SIN(PSI)
      COSPSI=COS(PSI)
*
*  COMPUTE REQUIRED FUNCTIONS OF EULER ANGLES - SCALE
*
*         -5G SIN(THETA)
      GSINTH=-1.61*SINTHE
*        G COS(PHI) COS(THETA)
      GCOSPH=.322*COSPHI*COSTHE
*         5G SIN(PHI) COS(THETA)
      GSINPH=1.61*SINPHI*COSTHE
*
```

```
*  CØØRDINATED TURN SYSTEM - DETERMINE TAIL RØTØR INPUT TØ ANALØG
*
      DA1C=DA1C*.5
*     CHECK CØØRDINATED TURN SWITCH
      IF(TEST(3).LT.0) GØ TØ 150
      DTHR=ABS(DTHR)
      IF(DTHR.GT..003) CALL SETLINES(1,1) ; GØ TØ 140
      CALL SETLINES(1,-1)
  140 DTHRC=0.
      GØ TØ 155
  150 CALL SETLINES(1,-1)
      BETA=ATAN(V/U)
      DTHRC=16.*DA1C+10.*BETA-3.*PSIDØT
*
*  CØMPUTE EULER ANGLE RATES - SCALE FØR ANALØG INPUT
*
  155 THEDØT=Q*CØSPHI-R*SINPHI
      PSIDØT=(Q*SINPHI+R*CØSPHI)/CØSTHE
      PHIDØT=P+PSIDØT*SINTHE
      THEDTS=-THEDØT*1.862
      PSIDTS= PSIDØT*28.65
      PHIDTS=-PHIDØT*1.91
*
*
*  CØMPUTE INERTIAL VELØCITIES - SCALE VZ FØR ANALØG INPUT
*
      V1=U*CØSTHE+(V*SINPHI+W*CØSPHI)*SINTHE
      V2=V*CØSPHI-W*SINPHI
      VX=V1*CØSPSI-V2*SINPSI
      VY=V1*SINPSI+V2*CØSPSI
      VZ=-U*SINTHE+V*SINPHI*CØSTHE+W*CØSPHI*CØSTHE
      VZS=VZ*.05
*
*  AIRSPEED INDICATØR - CØMPUTE SIGNAL FØR CØCKPIT INSTRUMENT
*
```

```
      IF(UKTS.GT.60.) GO TO 160
      ASI=.01*(59.-.4765*UKTS)
      GO TO 180
  160 IF(UKTS.GT.90.) GO TO 170
      ASI=.01*(100.-(10./9.)*UKTS)
      GO TO 180
  170 ASI=.01*(163.6-(100./55.)*UKTS)
*
*  RADAR ALTIMETER - COMPUTE SIGNAL FOR COCKPIT INSTRUMENT
*
  180 IF(ZE.GT.200.) RADALT=-.65                        ; GO TO 190
      IF(ZE.GT.100.) RADALT=-.40-.001983*(ZE-100.)  ; GO TO 190
      IF(ZE.GT.70.)  RADALT=-.345-.00183*(ZE-70.)   ; GO TO 190
      IF(ZE.GT.50.)  RADALT=-.300-.00225*(ZE-50.)   ; GO TO 190
      IF(ZE.GT.40.)  RADALT=-.2893-.00107*(ZE-40.)  ; GO TO 190
      IF(ZE.GT.20.)  RADALT=-.2429-.00232*(ZE-20.)  ; GO TO 190
      IF(ZE.GE.0.)   RADALT=-.2129-.0015*ZE         ; GO TO 190
      IF(ZE.LT.0.)   RADALT=-.2129
*
*  COMPUTE INERTIAL POSITION AND FLIGHT TIME
*
  190 CALL READCLOCK(TNEW)
      DT=.0001*(TNEW-TOLD)
      TOLD=TNEW
*     XE AND YE IN YARDS, ZE IN FEET
      XE=XE+VX*DT/3.
      YE=YE+VY*DT/3.
      ZE=ZE-VZ*DT
      FLTIM=.0001*TNEW
*
*  ACCUMULATE DATA TO COMPUTE RMS INERTIAL VELOCITIES AND POSITIONS
*
      IF(KFLAG.EQ.0) GO TO 195
      RMSTIM=RMSTIM+DT
      VXRMS=VXRMS+VX*VX*DT
```

```
      VYRMS=VYRMS+VY*VY*DT
      VZRMS=VZRMS+VZ*VZ*DT
      XERMS=XERMS+XE*XE*DT
      YERMS=YERMS+YE*YE*DT
      ZERMS=ZERMS+ZE*ZE*DT
*
*  INSTRUMENT DISPLAY DYNAMICS
*
  195 CALL INST
*
*  CREW DIRECTIONS
*
      IF(XE.LE.2000..AND.ZE.LE.250.) CALL CREW
*
*  PERFORM D-A AND A-D CONVERSIONS
*
      CALL DAC(1,VZS, 2,XAU, 3,ZAU, 4,MAU, 5,THEDTS, 6,PSIDTS, 7,PHIDTS,
     1          8,GSINTH, 9,GCOSPH, 10,GSINPH, 11,DTHRC, 12,ASI,
     2          16,MB1CU, 17,RADALT, 18,ZWU, 19,XTHCU, 20,ZB1CU, 21,LVU,
     3          22,NVU, 23,NRU)
      CALL ADK(C,PHI, 2,U, 3,THETA, 4,Q, 5,P, 6,R, 7,V, 8,W, 9,DA1C,
     1          10,DTHR)
*
*  SCALE VARIABLES AS REQUIRED
*
      U=U*250.
      UKTS=U/1.687
      V=V*50.
      W=W*50.
      P=P*.5
      Q=Q*.5
      R=R*.5
      PHI=PHI/1.91
      THETA=THETA/1.862+THETIC
      PSI=PSI+PSIDOT*DT
```

```
*
*  CHECK FØR STØP ØR CØNTINUE SIGNAL
*
*     IF STØP SWITCH ENERGIZED - EXIT DYNAMIC LØØP
      IF(TEST(2).LT.0) GØ TØ 210
*     IF FLY SWITCH ENERGIZED - CØNTINUE DYNAMIC LØØP
  200 IF(TEST(1).LT.0) IFLAG=IFLAG+1 ; GØ TØ 100
      GØ TØ 200
  210 CALL STØPCLØCK
      CALL HØLD
      RETURN
      END
```

```
      SUBRØUTINE INST
C*****************************************************************
C                                                                *
C  GENERATES DYNAMIC PØRTIØN ØF INSTRUMENT DISPLAY               *
C                                                                *
C*****************************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      CØMMØN DRVTAB(10,7),CRWDIR(7,17),B
      CØMMØN /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      CØMMØN /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDØT,
     1               PHIDØT,PSIDØT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      CØMMØN /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,CØSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTØP,YBØT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DIMENSIØN ICØMP(5),IVSIP(5),IDVIL(4),IALTP(5),ICMPS(38),IBANK(5),
     1          IHBAR(2),IPBAR(2),ITURNI(5),IBALL(5),ISPDP(5),IHDG(5)
      EQUIVALENCE (IVSIP(1),INDYM(2)), (IDVIL(1),INDYM(7)), (IALTP(1),
     1            INDYM(11)), (ICMPS(1),INDYM(16)), (IHDG(1),INDYM(54)),
     2            (IBANK(1),INDYM(59)), (IHBAR(1),INDYM(64)), (IPBAR(1),
     3            INDYM(66)), (ITURNI(1),INDYM(68)), (IBALL(1),
     4            INDYM(73)), (ISPDP(1),INDYM(78))
C
C  VERTICAL SPEED INDICATØR - PØINTER
C
      Y=YOVSI-DVSI*(AMAX(AMIN(VZ,16.7),-16.7))/4.17
      IVSIP(1)=IPACK(XPVSI,Y,0)
      X=XPVSI+HGT
      Y=Y-BASE/2.
      IVSIP(2)=IPACK(X,Y,1)
      Y=Y+BASE
      IVSIP(3)=IPACK(X,Y,1)
      Y=Y-BASE/2.
```

```
      IVSIP(4)=IPACK(XPVSI,Y,1)
      IVSIP(5)=IPACK(X,Y,1)
*
*  DIRECTION VELOCITY INDICATOR - SPEED INDICATOR LINES
*
      Y=YODVI-DDVI*(AMAX(AMIN(VX,23.7),-23.7))/5.93
      IDVIL(1)=IPACK(XLEFT,Y,0)
      IDVIL(2)=IPACK(XRIGHT,Y,1)
      X=XODVI-DDVI*(AMAX(AMIN(VY,23.7),-23.7))/5.93
      IDVIL(3)=IPACK(X,YTOP,0)
      IDVIL(4)=IPACK(X,YBOT,1)
*
*  CHECK FOR DESIRED DISPLAY
*
      IF(TEST(6).GE.0) NBLK=3 ; GO TO 400
      NBLK=9
*
*  RADAR ALTIMETER - POINTER
*
      IF(ZE.GT.100.) GO TO 200
      Y=YOALT+DALT*(AMAX(ZE,-10.))/5.
      GO TO 210
  200 IF(ZE.GT.200.) GO TO 205
      Y=YOALT+DALT*(20.+(ZE-100.)/25.)
      GO TO 210
  205 Y=YOALT+DALT*(24.+(AMIN(ZE,550.)-200.)/100.)
  210 IALTP(1)=IPACK(XPALT,Y,0)
      X=XPALT+HGT
      Y=Y-BASE/2.
      IALTP(2)=IPACK(X,Y,1)
      Y=Y+BASE
      IALTP(3)=IPACK(X,Y,1)
      Y=Y-BASE/2.
      IALTP(4)=IPACK(XPALT,Y,1)
      IALTP(5)=IPACK(X,Y,1)
```

```
*
*  CØMPASS HEADING
*
      PSIDEG=PSI*57.3
      IF(PSIDEG.LE.0.) PSIDEG=PSIDEG+360.
      IPSI=5*(INT(PSIDEG/5.+.5))
      K=(72+IPSI)/360
*     IF HEADING REMAINS CØNSTANT DØ NØT REGENERATE TEXT
      IF(K.EQ.KSAV) GØ TØ 250
      KSAV=K
      ENCØDE(20,345,ICØMP)(CMPNUM(J),J=K,K+18)
  345 FØRMAT(19A1)
      CALL TEXTB(IDEV,ICØMP,5,5,30,2,3,IER)
*
*  CØMPASS - SCALE
*
      L=2*(K/2)
      D1=.10*SCALE
      D2=.25*SCALE
      IF(K.NE.L) D1=.25*SCALE ; D2=.10*SCALE
      X=-.45
      Y=.382
      ICMPS(1)=IPACK(X,Y,0)
      DØ 240 J=2,34,4
        Y=Y+D1
        ICMPS(J)=IPACK(X,Y,1)
        X=X+DCMPS
        Y=Y-D1
        ICMPS(J+1)=IPACK(X,Y,0)
        Y=Y+D2
        ICMPS(J+2)=IPACK(X,Y,1)
        X=X+DCMPS
        Y=Y-D2
        ICMPS(J+3)=IPACK(X,Y,0)
  240   CØNTINUE
```

```
      Y=Y+D1
      ICMPS(38)=IPACK(X,Y,1)
*
*  COMPASS - HEADING POINTER
*
  250 XOHDG=(PSIDEG-IPSI)*DCMPS/2.5
      IHDG(1)=IPACK(XOHDG,YOHDG,0)
      X=XOHDG-BASE/2.
      Y=YOHDG-HGT
      IHDG(2)=IPACK(X,Y,1)
      X=X+BASE
      IHDG(3)=IPACK(X,Y,1)
      IHDG(4)=IPACK(XOHDG,YOHDG,1)
      IHDG(5)=IPACK(XOHDG,Y,1)
*
*  VERTICAL GYRO INDICATOR - ANGLE OF BANK POINTER
*
      XP=XCEN-RP*SINPHI
      YP=YCEN+RP*COSPHI
      IBANK(1)=IPACK(XP,YP,0)
      X=XCEN-RB*SIN(PHI+EPS)
      Y=YCEN+RB*COS(PHI+EPS)
      IBANK(2)=IPACK(X,Y,1)
      X=XCEN-RB*SIN(PHI-EPS)
      Y=YCEN+RB*COS(PHI-EPS)
      IBANK(3)=IPACK(X,Y,1)
      IBANK(4)=IPACK(XP,YP,1)
      X=XCEN-RB*SINPHI
      Y=YCEN+RB*COSPHI
      IBANK(5)=IPACK(X,Y,1)
*
*  VERTICAL GYRO INDICATOR - HORIZON BAR
*
      PITCH=-(AMAX(AMIN(THETA,.87),-.87))
      RC=PITCH*ASCALE
```

```
      YO=YCEN+RO*COSPHI
      RO=ABS(RO)
      XO=XCEN-RO*SINPHI
      DL=SQRT(RB*RB-RO*RO)
      X=XO-DL*COSPHI
      Y=YO-DL*SINPHI
      IHBAR(1)=IPACK(X,Y,0)
      X=XO+DL*COSPHI
      Y=YO+DL*SINPHI
      IHBAR(2)=IPACK(X,Y,1)
*
*  VERTICAL GYRO INDICATOR - PITCH BAR
*
      RO=(PITCH+.1745)*ASCALE
      YO=YCEN+RO*COSPHI
      RO=ABS(RO)
      XO=XCEN+RO*SINPHI
      DL=.80*SCALE
      X=XO-DL*COSPHI
      Y=YO-DL*SINPHI
      IPBAR(1)=IPACK(X,Y,0)
      X=XO+DL*COSPHI
      Y=YO+DL*SINPHI
      IPBAR(2)=IPACK(X,Y,1)
*
*  RATE OF TURN - INDICATOR
*
      DS=.04
      DL=.08
      Y=YOTURN-.016
      X=XOTURN-DS/2.+DTURN*(AMAX(AMIN(PSIDOT,.105),-.105))/.0523
      ITURNI(1)=IPACK(X,Y,0)
      X=X+DS
      ITURNI(2)=IPACK(X,Y,1)
      Y=Y-DL
```

```
      ITURNI(3)=IPACK(X,Y,1)
      X=X-DS
      ITURNI(4)=IPACK(X,Y,1)
      Y=Y+DL
      ITURNI(5)=IPACK(X,Y,1)
*
*  SLIP INDICATØR - BALL
*
      DS=.18*SCALE
      X=XOSLIP-DS/2.+.862*(PHI-ATAN(U*PSIDØT/32.2))
      Y=YOSLIP+DS/2.
      IBALL(1)=IPACK(X,Y,0)
      X=X+DS
      IBALL(2)=IPACK(X,Y,1)
      Y=Y-DS
      IBALL(3)=IPACK(X,Y,1)
      X=X-DS
      IBALL(4)=IPACK(X,Y,1)
      Y=Y+DS
      IBALL(5)=IPACK(X,Y,1)
*
*  AIRSPEED - SCALE PØINTER
*
      UK=AMAX(AMIN(UKTS,105.),0.)
      IF(UK.GT.20.) GØ TØ 300
      Y=YOSPD+DSPD*UK/10.
      GØ TØ 305
  300 IF(UK.GT.100.) UK=100.
      Y=YOSPD+DSPD*(2.+(UK-20.)/5.)
  305 ISPDP(1)=IPACK(XPSPD,Y,0)
      X=XPSPD-HGT
      Y=Y-BASE/2.
      ISPDP(2)=IPACK(X,Y,1)
      Y=Y+BASE
      ISPDP(3)=IPACK(X,Y,1)
```

```
      Y=Y-BASE/2.
      ISPDP(4)=IPACK(XPSPD,Y,1)
      ISPDP(5)=IPACK(X,Y,1)
      GØ TØ 420
  400 DØ 410 I=11,82
        INDYM(I)=0
  410   CØNTINUE
  420 CALL GRAPHØ(IDEV,INDYM,83,NBLK,IER)
      IF(IER.NE.0) ØUTPUT(6) 'ERRØR - INDYM',IER
      RETURN
      END
```

```
      SUBR0UTINE CREW
*****************************************************
*                                                   *
*  GENERATE APPR0PIRATE DIRECTI0NS FR0M CREW        *
*                                                   *
*****************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      C0MM0N DRVTAB(10,7),CRWDIR(7,17),B
      C0MM0N /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90)
      C0MM0N /STATE/ U,V,W,JKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THED0T,
     1               PHID0T,PSID0T,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      AXE=ABS(XE)
      IF(XE.GE.-1100.) G0 T0 705
      I=17
      J=1
      IX=NULL
      IY=NULL
      H0VTIM=0.
      G0 T0 765
  705 IF(XE.GE.-500.) G0 T0 710
      I=1
      J=16
      AXE=200.*INT(AXE/200.+.5)
      ENCODE(4,730,IX) AXE
      H0VTIM=0.
      G0 T0 765
  710 IF(J.EQ.13) G0 T0 755
      IF(J.EQ.14) G0 T0 765
      IF(AXE.LE.3.)  I=4 ; IX=NULL ; G0 T0 735
      IF(XE.GE.3.)   G0 T0 715
      IF(XE.LT.-30.) I=2 ; G0 T0 720
      IF(VX.LT.B*AXE) I=2 ; G0 T0 720
      I=3
```

```
      IX=NULL
      GØ TØ 735
  715 IF(XE.GT.60.)  I=7 ; J=8 ; IX=NULL ; IY=NULL ; GØ TØ 765
      IF(XE.GT.20.)  I=6 ; GØ TØ 720
      IF(VX.GT.-8*AXE) I=6 ; GØ TØ 720
      I=5
      IX=NULL
      GØ TØ 735
  720 IF(AXE.GE.100.) AXE=100.*INT(AXE/100.+.5) ; GØ TØ 725
      IF(AXE.GT.10.)  AXE=10.*INT(AXE/10.+.5)   ; GØ TØ 725
      AXE=INT(AXE+.5)
  725 ENCØDE(4,730,IX) AXE
  730 FØRMAT(I4)
*
*  CHECK LATERAL PØSITIØN
*
  735 AYE=ABS(YE)
      IF(AYE.LE.3.)   I=16 ; IY=NULL ; GØ TØ 752
      IF(I.EQ.4)      I=16
      IF(YE.LT.-3.)   GØ TØ 740
      IF(YE.GT.25.)   J=9 ; GØ TØ 745
      IF(VY.GT.-8*AYE) J=9 ; GØ TØ 745
      J=10
      IY=NULL
      GØ TØ 752
  740 IF(YE.LT.-25.) J=11 ; GØ TØ 745
      IF(VY.LT.8*AYE) J=11 ; GØ TØ 745
      J=12
      IY=NULL
  745 IF(AYE.GT.10.) AYE=10.*INT(AYE/10.+.5) ; GØ TØ 747
      AYE=INT(AYE+.5)
  747 ENCØDE(4,730,IY) AYE
*
*  CØMPUTE HØVER TIME
*
```

```
752 IF(JFLAG.EQ.1) GØ TØ 754
753 IF(I.EQ.4.AND.J.EQ.16) JFLAG=1 ; KFLAG=1 ; GØ TØ 755
    HØVTIM=0.
    GØ TØ 765
754 IF(AXE.GT.10..ØR.AYE.GT.10.) JFLAG=0 ; HØVTIM=0. ; GØ TØ 765
755 HØVTIM=HØVTIM+DT
    IF(HØVTIM.GT.150.) I=4 ; J=14 ; KFLAG=0 ; GØ TØ 765
    IF(HØVTIM.GT.120.) I=4 ; J=13 ; GØ TØ 765
*
*  CHECK ALTITUDE
*
765 IF(ZE.LE.15.) I=15 ; J=16 ; GØ TØ 775
*
*  ØUTPUT CREW DIRECTIØNS
*
770 CRWDIR(5,I)=IX
    CRWDIR(5,J)=IY
775 CALL TEXTØ(IDEV,CRWDIR(1,I),6,35,29,2,3,IER)
    CALL TEXTØ(IDEV,CRWDIR(1,J),6,37,29,2,3,IER)
    RETURN
    END
```

STABILITY DERIVATIVES

	0 KTS	30 KTS	50 KTS	70 KTS	91 KTS	112 KTS	136 KTS
XAU	.0000	.2328	-1.5014	-3.8143	-6.8470	-10.6184	-15.5095
ZAU	.0000	-9.2718	-5.8539	.6023	6.5540	18.7617	34.3457
MAU	.0000	.1417	.1940	.2244	.2610	.2456	.1835
ZAU	-.4045	-.5092	-.5843	-.6368	-.6682	-.6875	-.6928
YB1CU	-12.1694	-12.2229	-12.3438	-12.5480	-12.8240	-13.0365	-13.1535
LVU	-.0215	-.0261	-.0298	-.0352	-.0409	-.0464	-.0519
NRU	-.5371	-.7410	-.8881	-1.0799	-1.2692	-1.4471	-1.6220
NVU	.0172	.0202	.0227	.0272	.0312	.0352	.0399
XTHCU	25.0600	22.8800	21.9600	19.9420	17.5500	17.6170	20.4540
ZB1CU	4.5674	35.3516	64.7892	97.4090	131.0581	164.4026	197.3886

CREW DIRECTIØNS

STEADY FØRWARD	EASY FØRWARD	STØP FØRWARD
STEADY HØVER	STØP BACK	EASY BACK
TARGET LØST	WAVE ØFF	EASY LEFT
STØP LEFT	EASY RIGHT	STØP RIGHT
MAN ØN HØIST	MAN IN AIRCRAFT	PULL UP - YØU ARE LØW
	TARGET IN SIGHT	

ALTITUDE SCALE NUMBERS

0 20 40 60 80 100 150 200 500

AIRSPEED SCALE NUMBERS

0 20 30 40 50 60 70 80 90 100

CØMPASS NUMBERS

330	360	030	060	090	120	150	180
210	240	270	300	330	360	030	

APPENDIX B

DIGITAL COMPUTER PROGRAM FORTRAN VARIABLES

ABS	Absolute value -- intrinsic subprogram.
ADK	External subprogram used to perform analog to digital conversion.
AIRNUM	Airspeed numbers -- array containing the numbers for the airspeed scale.
ALTNUM	Altimeter numbers -- array containing the numbers for the radar altimeter scale.
AMAX	Maximum value of two arguments -- intrinsic subprogram.
AMIN	Minimum value of two arguments -- intrinsic subprogram.
AROLL	Absolute value of ROLL.
ASCALE	Angle scale -- scale factor for converting an angle to a linear displacement.
ASI	Airspeed indicator -- scaled value of airspeed sent to cockpit indicator.
ATAN	Arctangent -- intrinsic subprogram.
AXE	Absolute value of XE.
AYE	Absolute value of YE.
BASE	Base -- length of the base of the triangular pointers used in the integrated display.
BETA	Sideslip angle.
CMPNUM	Compass numbers -- array containing the numbers for the compass scale.
COMPUTE	External subprogram used to place the analog computer in the "compute" mode.
COS	Cosine -- intrinsic subprogram.

COSPHI — Cosine of PHI.

COSPSI — Cosine of PSI.

COSTHE — Cosine of THETA.

CREW — Crew -- subprogram which generates directions from a simulated rescue aircrewman.

CRWDIR — Crew directions -- array containing crew directions.

D1 — Length of a scale mark on the compass heading scale.

D2 — Length of a scale mark on the compass heading scale.

DA1C — ΔA_{1c}

DAC — External subprogram used to perform digital to analog conversion.

DALT — Altimeter division -- distance between divisions of the radar altimeter scale.

DCMPS — Compass division -- distance between divisions of the compass heading scale.

DDVI — Direction velocity indicator division -- distance between divisions of the direction velocity indicator scale.

DGINIT — Graphics initialization subroutine.

DL — Long displacement -- length of a long scale mark.

DRVNAM — Derivative name -- array containing the names of the stability derivatives.

DRV — Derivative -- array containing the airspeed dependent stability derivatives for a specified airspeed.

DRVTAB — Derivative table -- array containing the airspeed dependent stability derivatives for several airspeeds.

DS — Short displacement -- length of a short scale mark.

DSPD — Speed division -- distance between divisions of the airspeed scale.

DSPLY — Display -- subprogram which generates the static portions of the integrated instrument display.

DT — Time interval.

DTHR $\Delta\theta_R$

DTHRC Value of $\Delta\theta_R$ required to maintain zero sideslip flight.

DTINIT Text initialization subroutine.

DTURN Turn division -- distance between marks of the turn indicator.

DVSI Vertical speed indicator division -- distance between divisions of the vertical speed indicator.

EPS Small angle.

FLTIM Flight time.

FLY Fly -- subprogram which generates information for and controls the solution of the helicopter dynamics.

GCOSPH Factor in Z-Force equation.

GRAPHO Graphics output -- external subprogram, used to output a graphics array to the graphics processor.

GSINPH Factor in Y-Force equation.

GSINTH Factor in X-Force equation.

HGT Height -- height of the triangular pointers used in the integrated display.

HOLD External subprogram used to place the analog computer in the "hold" mode.

HOVTIM Hover time -- elapsed time within a specified distance from the target.

I Integer counter.

IALT Altimeter -- graphics data array for the radar altimeter scale.

IALTP Altimeter pointer -- graphics data array for the radar altimeter pointer.

IBALL	Ball -- graphics data array for the slip indicator ball.
IBANK	Bank angle -- graphics data array for the attitude gyro angle of bank pointer.
IBLANK	Blank -- graphics data array used to blank out another graphics data array.
ICMPS	Compass -- graphics data array for the compass heading scale.
ICOMP	Compass -- text array for the compass heading numbers.
ICOPS	Initial character position -- fixes the lateral position on the graphics display of the first character in a text array.
IDEV	Device number -- the number 1 or 2 which specifies the graphics processor to be used.
IDVI	Direction velocity indicator -- graphics data array for the direction velocity indicator scales.
IDVIL	Direction velocity indicator lines -- graphics data array for the direction velocity indicator speed lines.
IER	Error parameter returned after calls to DGINIT, DTINIT, GRAPHO or TEXTO.
IFLAG	Integer counter -- counts number of times through dynamic loop in FLY.
IGD	Graphics directory -- argument of DGINIT.
IHBAR	Horizon bar -- graphics data array for the attitude gyro artificial horizon line.
IHEAD	External subprogram used to generate the first word of a graphics array.
IHDG	Heading -- graphics data array for the compass heading pointer.
IMAP	Miniature airplane -- graphics data array for the attitude gyro miniature airplane reference.
INDYM	Instrument dynamics -- graphics data array for the moving (dynamic) portions of the instrument display.
INT	Converts a number to an integer -- intrinsic subprogram.
INST	Instrument -- subprogram which generates the dynamic portions of the integrated display.

IPACK	External subprogram used to generate words of a graphics array.
IPSI	PSI converted to integer value.
IPBAR	Pitch bar -- graphics data array for the attitude gyro pitch line.
ISLIP	Slip -- graphics data array for the slip indicator center marks.
ISPD	Speed -- graphics data array for the airspeed scale.
ISPDP	Speed pointer -- graphics data array for the airspeed pointer.
ITD	Text directory -- argument of DTINIT.
ITURN	Turn -- graphics data array for the turn indicator scale.
ITURNI	Turn indicator -- graphics data array for the turn needle.
IVGI	Vertical gyro indicator -- graphics data array for the attitude gyro angle of bank scale.
IVSI	Vertical speed indicator -- graphics data array for the vertical speed indicator scale.
IVSIP	Vertical speed indicator pointer -- graphics data array for the vertical speed indicator pointer.
IX	Integer X -- integer value of XE.
IY	Integer Y -- integer value of YE.
J	Integer counter.
JFLAG	Integer flag used to control accumulation of hover time.
K	Integer counter.
K2	Integer value retained for later comparison.
KFLAG	Integer flag used to control accumulation of RMS performance parameters.
KSAV	K save -- saves value of K for later comparison.

L	Integer counter.
LN	Line number -- specifies line position of a text block.
LNO	Same as LN except refers to initial line.
LVU	$L_v(u)$
MAU	$M_A(u)$
MB1CU	$M_{B_{1c}}(u)$
NBLK	Block number -- refers to graphics data blocks.
NGD	Number of words in the graphics directory -- argument of DGINIT.
NRU	$N_R(u)$
NTD	Number of words in the text directory -- argument of DTINIT.
NULL	Null -- text array of blank spaces.
NVU	$N_v(u)$
P	p
PHI	ϕ
PHIDOT	$\dot{\phi}$
PHIDTS	$\dot{\phi}$ scaled for the analog computer.
PITCH	θ limited to $\pm 50^o$.
POTSET	Subprogram which places the analog computer in the POTSET mode.
PSI	ψ
PSIDEG	ψ scaled to degrees.
PSIDOT	$\dot{\psi}$
PSIDTS	$\dot{\psi}$ scaled for the analog computer.

Q	q
R	r
RO	Initial radius -- radial distance from the center of the attitude gyro to the angle of bank scale marks.
RADALT	Radar altimeter -- scaled value of altitude sent to cockpit radar altimeter.
RB	Radius to base -- radial distance from the center of the attitude gyro to the base of the triangular angle of bank pointer.
READCLOCK	External subprogram used to read the present value of the analog computer clock.
RESET	Reset -- subprogram which places analog computer in Reset mode.
RL	Radial line -- length of the radial line segment used for the attitude gyro angle of bank scale marks.
RMSTIM	Root mean square time -- time interval used to compute performance parameters.
ROLL	Roll -- angular position of the attitude gyro angle of bank scale marks.
RP	Radius to point -- radial distance from the center of the attitude gyro to the point of the angle of bank pointer.
SETLINES	External subprogram used to set analog computer logic.
SETPOT	External subprogram used to set the analog computer potentiometers.
SCALE	Scale -- multiplying factor to convert ± 5 inches to ± 1 units for graphics processor.
SIN	Sine -- intrinsic subprogram.
SINPHI	Sine of PHI.
SINPSI	Sine of PSI.
SINTHE	Sine of THETA.
SQRT	Square root -- intrinsic subprogram.

STARTCLOCK	External subprogram used to start the analog computer clock.
STOPCLOCK	External subprogram used to stop the analog computer clock.
TEST	External subprogram used to test the logic of specified analog trunk lines.
TEXTO	External subprogram used to output a text array to the graphics processor.
THEDOT	$\dot{\theta}$
THEDTS	$\dot{\theta}$ scaled for the analog computer.
THETA	θ
THETIC	Initial condition on θ.
TNEW	New time.
TOLD	Old time.
U	u
U1	Value of airspeed used in the linear interpolation subroutine.
U2	Value of airspeed used in the linear interpolation subroutine.
UK	u scaled to knots.
UKTS	u scaled to knots.
V	v
V1	Intermediate calculation for VX and VY.
V2	Intermediate calculation for VX and VY.
VX	Inertial velocity along the x-axis.
VXRMS	Root mean square value of VX. Used as a performance parameter.

VY — Inertial velocity along the y-axis.

VYRMS — Root mean square value of VY. Used as a performance parameter.

VZ — Inertial velocity along the z-axis.

VZRMS — Root mean square value of VZ. Used as a performance parameter.

VZS — VZ scaled for the analog computer.

W — w

WRITECLOCK — External subprogram used to set the analog computer clock to a specified value.

X — x coordinate position used for graphics construction.

XO — X initial -- initial x coordinate position used for graphics construction.

XOALT — XO for the radar altimeter scale.

XODVI — XO for the direction velocity indicator scale.

XOHDG — XO for the compass heading pointer.

XOSLIP — XO for the slip indicator scale.

XOSPD — XO for the airspeed scale.

XOTURN — XO for the turn rate scale.

XOVSI — XO for the vertical speed scale.

XAU — $X_A(u)$

XCEN — x center -- x coordinate position for the center of the attitude gyro.

XE — x earth -- x coordinate position of the helicopter in the inertial reference axes.

XERMS — XE root mean square -- used as a performance parameter.

XLEFT — x left -- x coordinate position of the left end of the direction velocity indicator speed line.

XP	x pointer -- x coordinate position for the point of a scale pointer.
XPALT	XP for the radar altimeter scale pointer.
XPSPD	XP for the airspeed scale pointer.
XPVSI	XP for the vertical speed scale pointer.
XRIGHT	x right -- x coordinate position for the right end of the direction velocity indicator speed line.
XTHCU	$X_{\theta_c}(u)$
Y	y coordinate position used for graphics construction.
YO	y initial -- initial x coordinate position used for graphics construction.
YOALT	YO for the radar altimeter scale.
YODVI	YO for the direction velocity indicator scale.
YOHDG	YO for the compass heading pointer.
YOSLIP	YO for the slip indicator scale.
YOSPD	YO for the airspeed scale.
YOTURN	YO for the turn rate scale.
YOVSI	YO for the vertical speed scale.
YBOT	y bottom -- y coordinate position for the bottom end of the direction velocity indicator speed line.
YCEN	y center -- y coordinate position for the center of the attitude gyro.
YE	y earth -- y coordinate position of the helicopter in the inertial reference axes.
YERMS	YE root mean square -- used as a performance parameter.
YP	y pointer -- y coordinate position for the point of a scale pointer.
YTOP	y top -- y coordinate position for the top end of the direction velocity indicator speed line.

ZAU	$Z_A(u)$
ZB1CU	$Z_{B_{1c}}(u)$
ZE	z earth -- z coordinate position for the helicopter in the inertial reference axes (altitude).
ZEIC	ZE initial condition -- starting value of ZE.
ZERMS	ZE root mean square -- used as a performance parameter.
ZES	ZE scaled for the analog computer.
ZWU	$Z_w(u)$

APPENDIX C

ANALOG COMPUTER PROGRAM

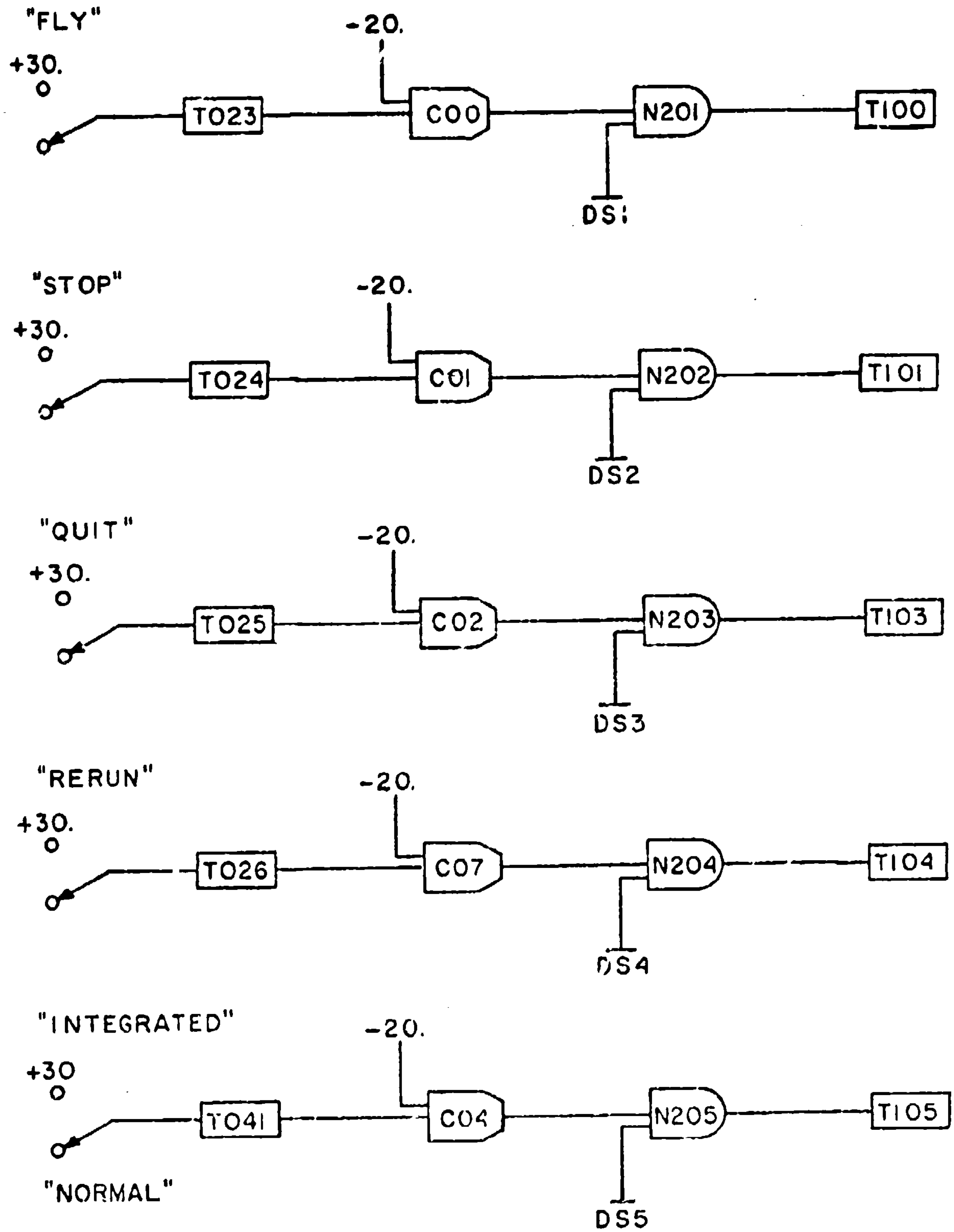
This appendix contains a listing of the trunk lines between the analog computer and the cockpit together with the signal carried; a listing of the analog computer potentiometer settings; and the patching diagrams for the analog computer program.

T000	$200\Delta\theta_c$
T001	$200\Delta A_{1c}$
T002	$100\Delta B_{1c}$
T003	+30 VDC
T004	-30 VDC
T005	AIRSPEED
T006	BALL
T007	TURN NEEDLE
T010	HEADING
T011	ALTIMETER
T014	RADAR ALTIMETER
T015	PITCH ATTITUDE
T016	ROLL ATTITUDE
T017	VERTICAL SPEED
T020	$500\Delta\theta_R$
T023	"FLY"
T024	"STOP"
T025	"QUIT"
T026	"RERUN"
T030	COORDINATED TURN
T041	"DISPLAY TYPE"

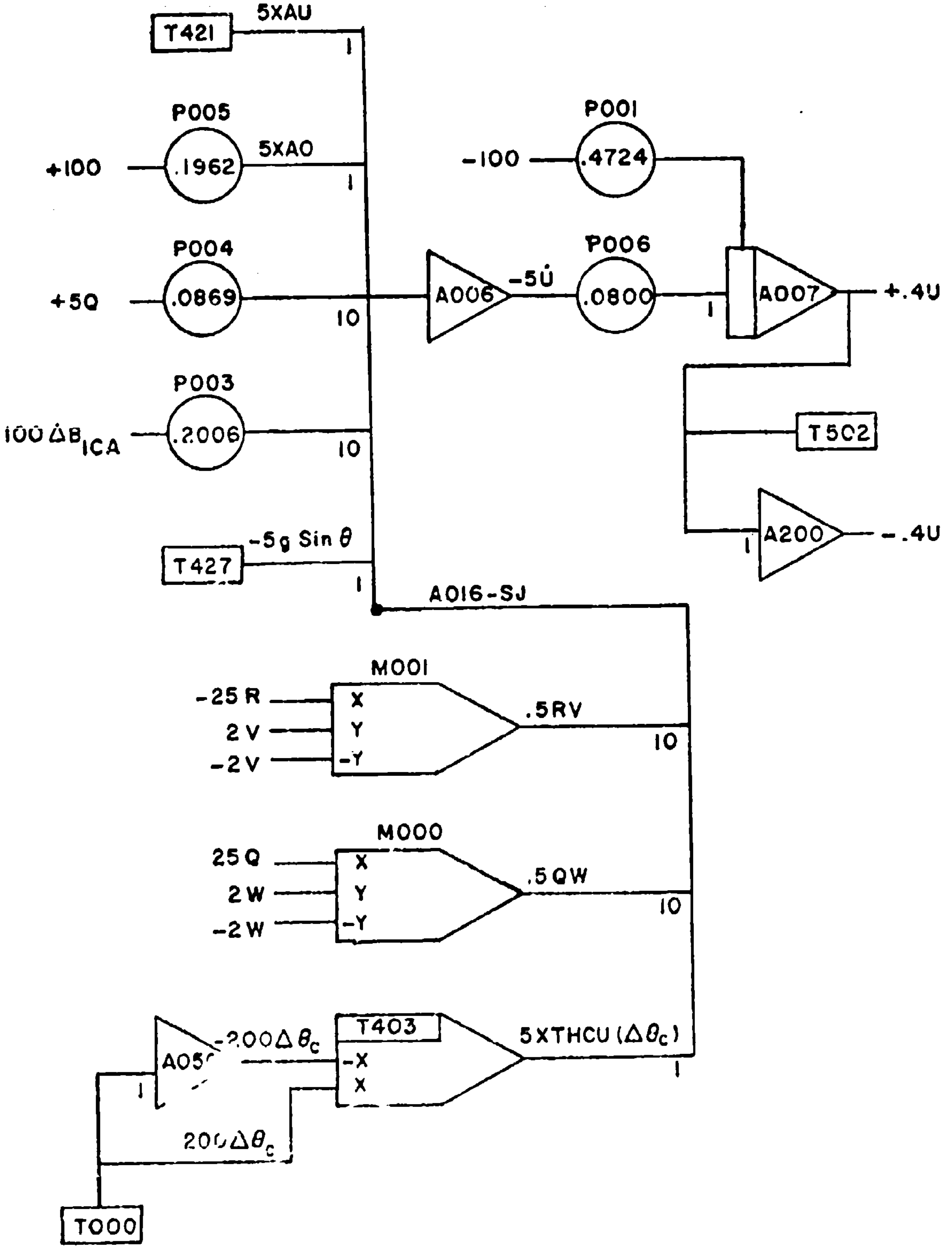
TABLE CI. -- USE OF TRUNK LINES

POT NUMBER	SETTING	POT NUMBER	SETTING
000	.0890	032	.9127
001	.4724	033	.1021
002	.1320	034	.6062
003	.2006	035	.2598
004	.0869	036	.4000
005	.1962	037	.2500
006	.0800	040	.1093
010	.0100	041	.0018
011	.1224	042	.4692
012	.2500	043	.1186
013	.1250	044	.4000
014	.0100	045	.1250
015	.0838	046	.6250
016	.0250	047	.2500
017	.7075	050	.3200
020	.7853	051	.1041
021	.0389	052	.2000
022	.1066	053	.1052
023	.0241	054	.1052
024	.2000	055	.6282
025	.5000	056	.5000
026	.0235	400	.2000
027	.0845	401	.2000
030	.4000	436	.3000
031	.1816	437	.3000

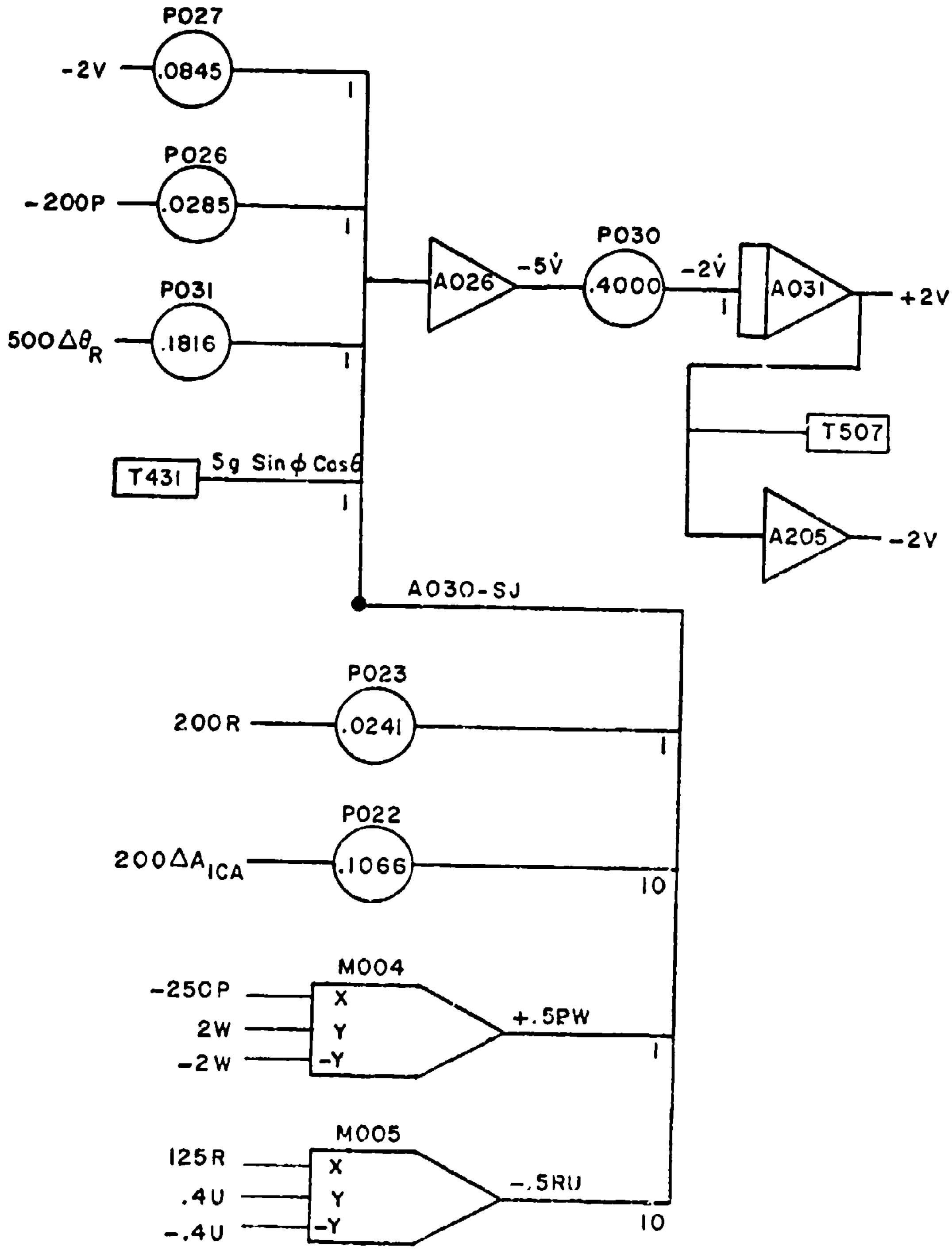
TABLE CII. -- ANALOG COMPUTER POTENTIOMETER SETTINGS



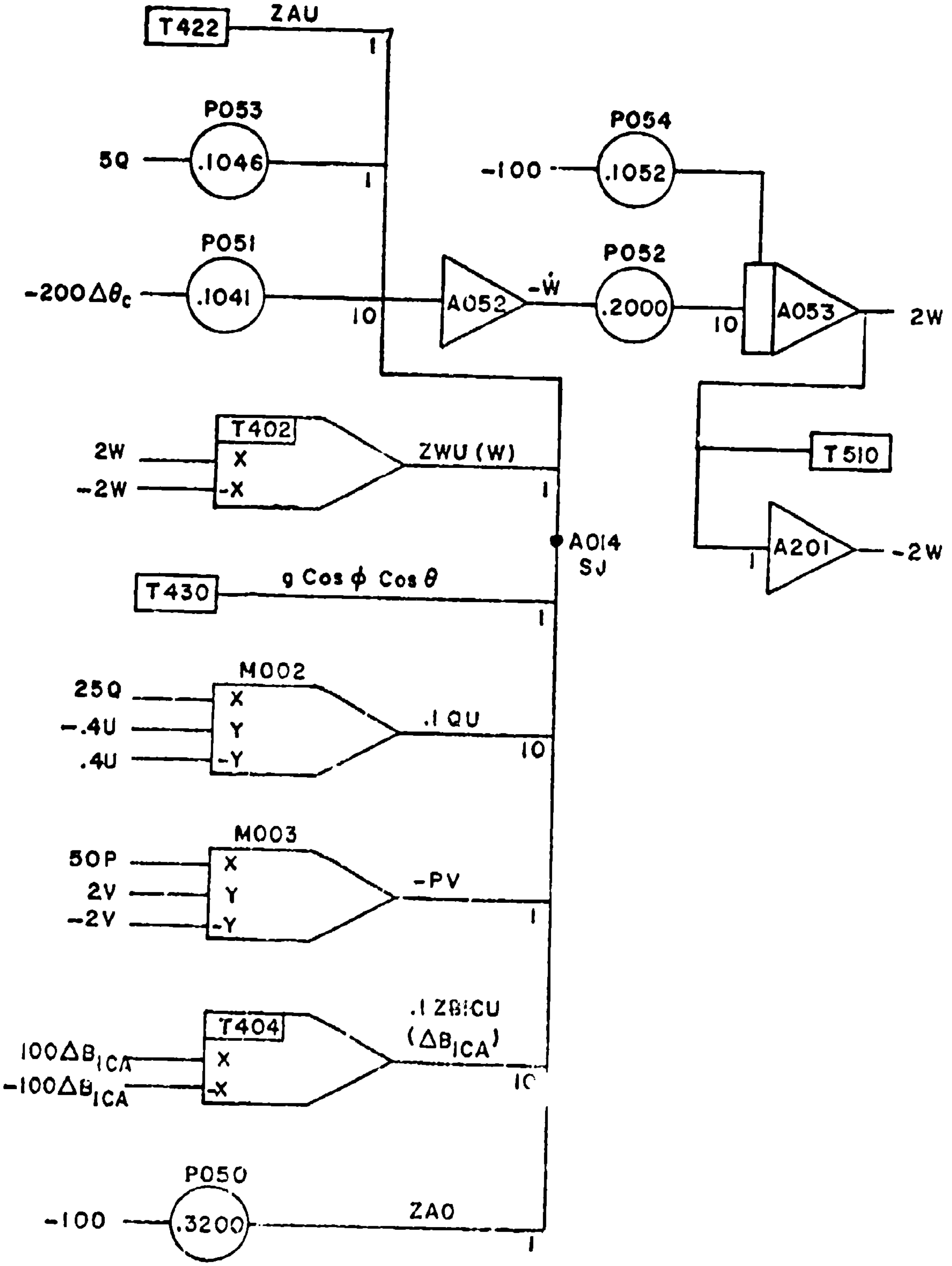
SIMULATION CONTROL



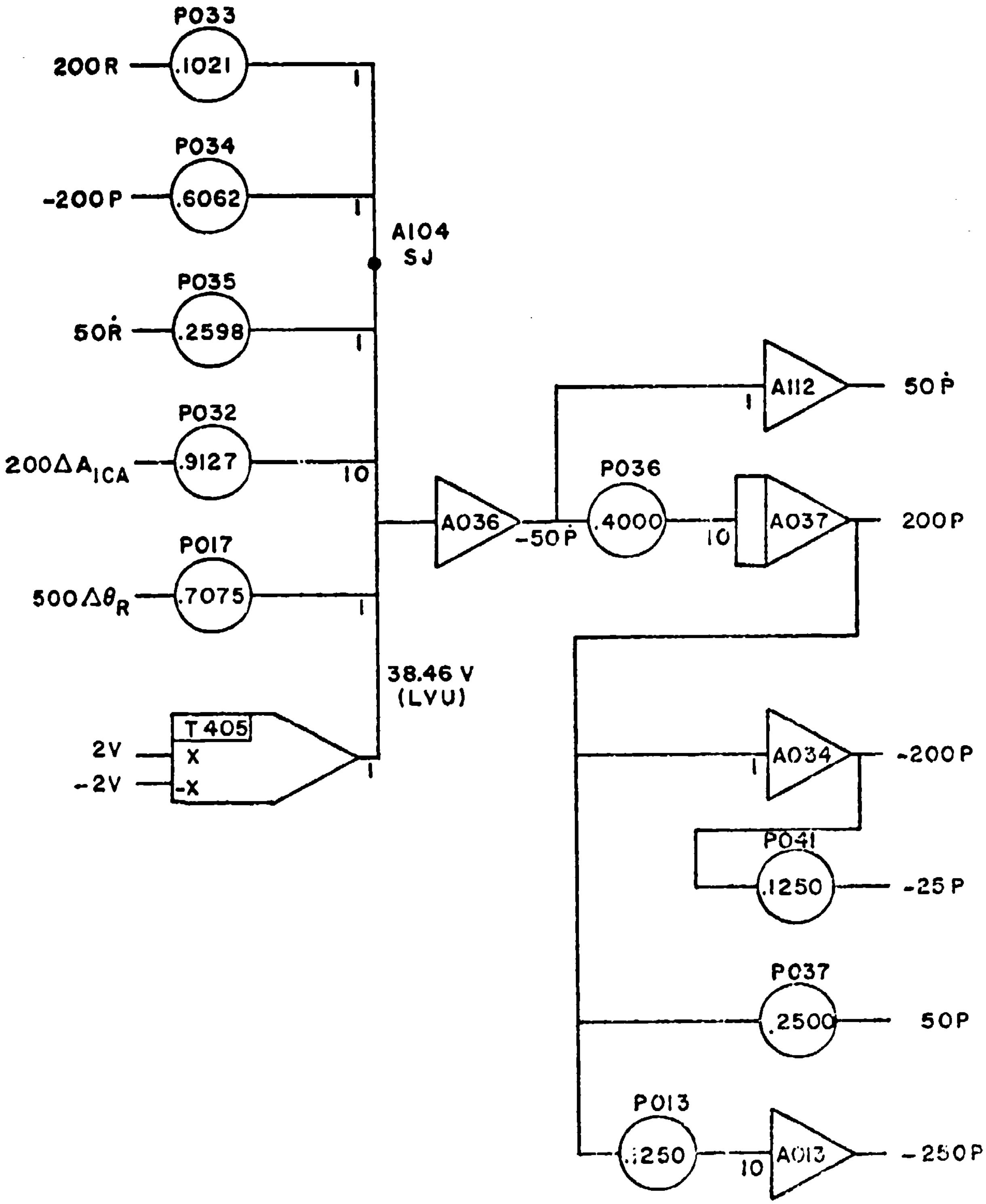
X FORCE EQUATION



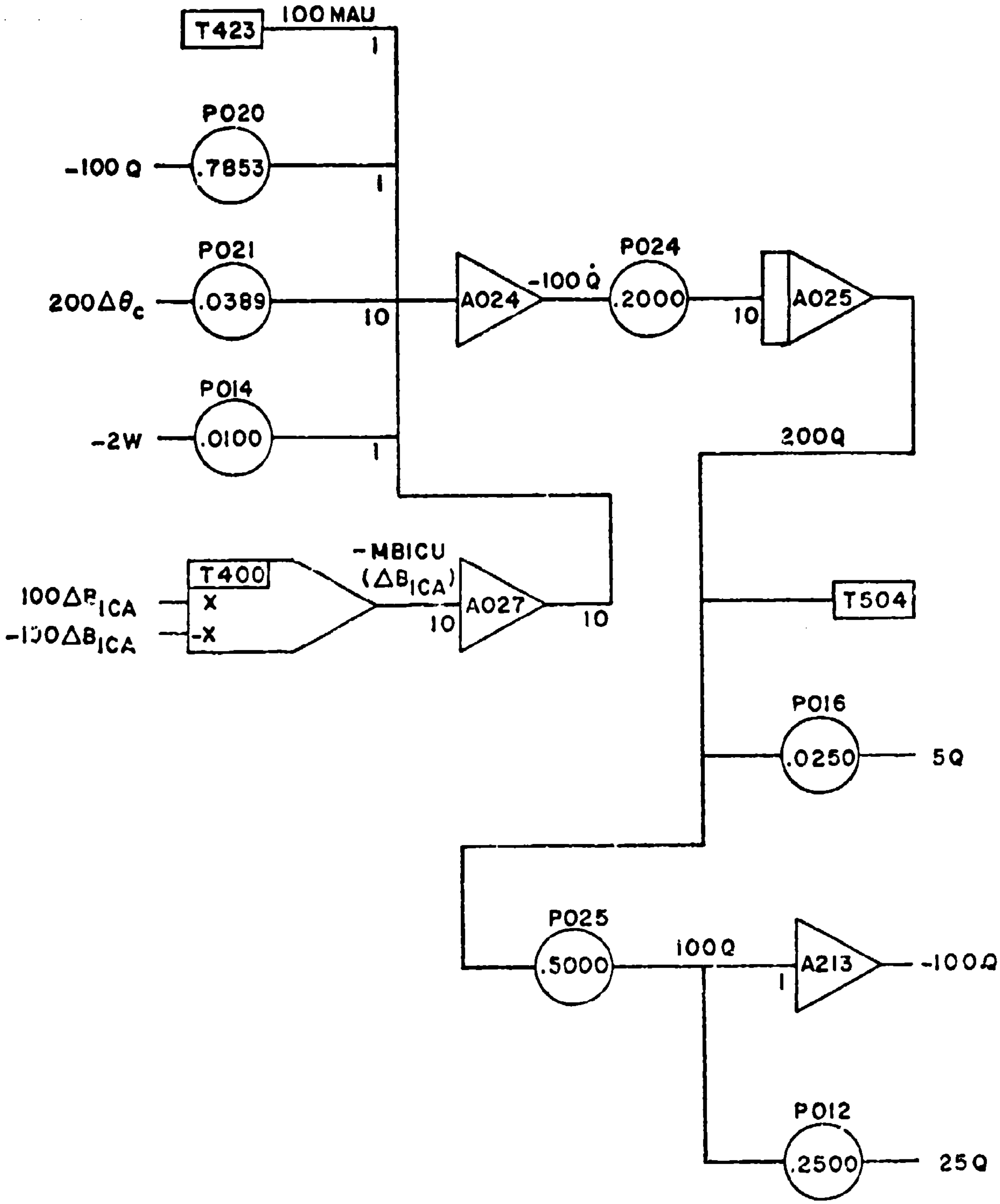
Y FORCE EQUATION



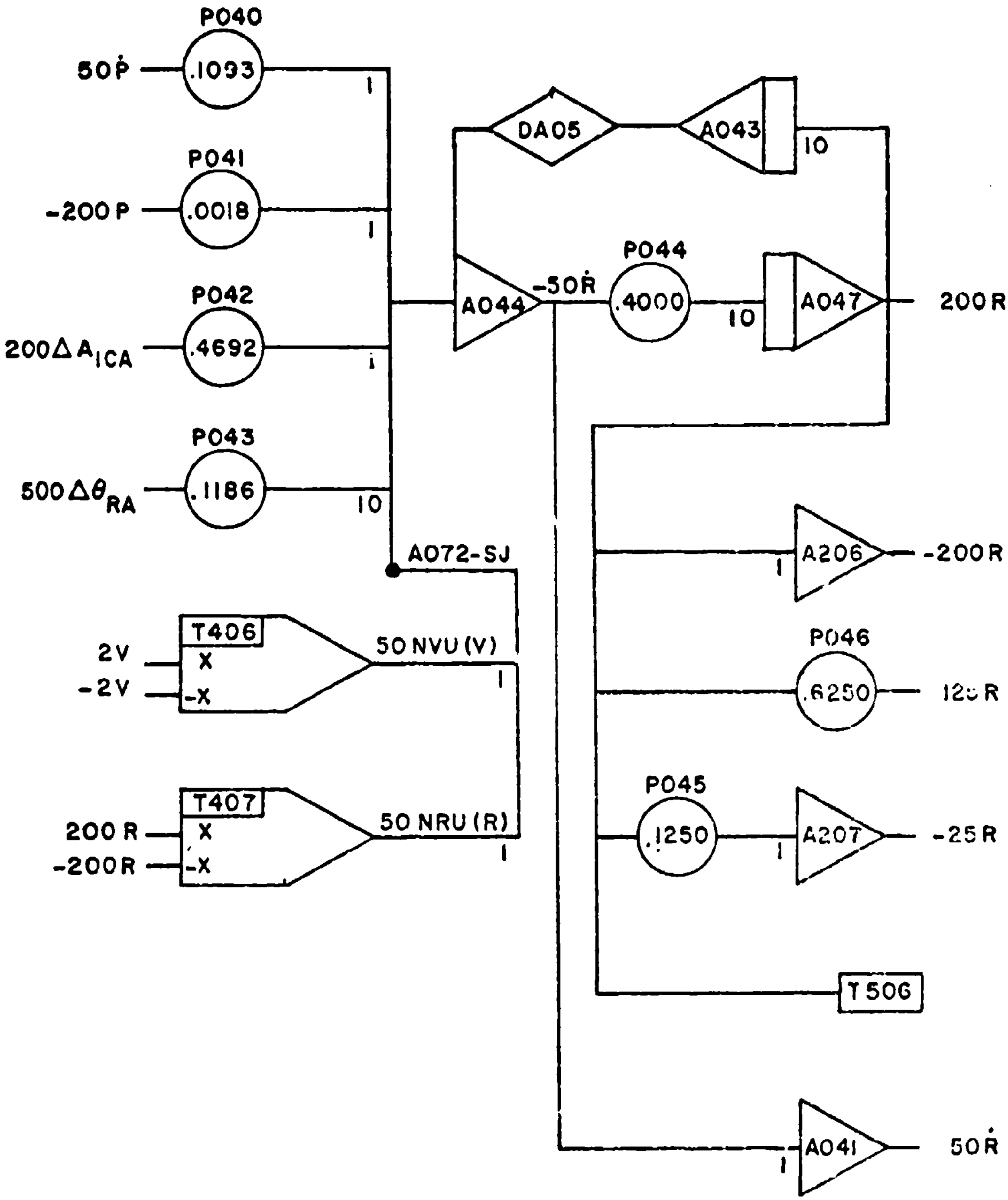
Z FORCE EQUATION



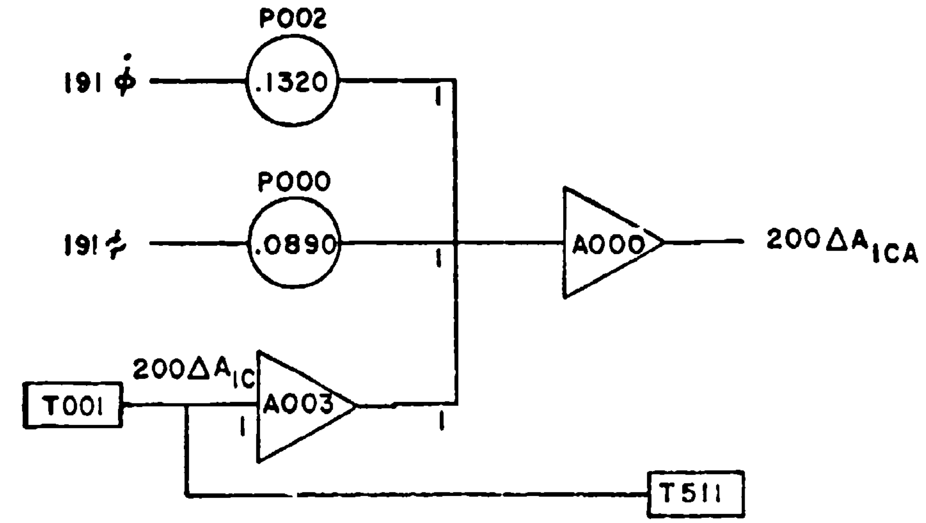
ROLL MOMENT EQUATION



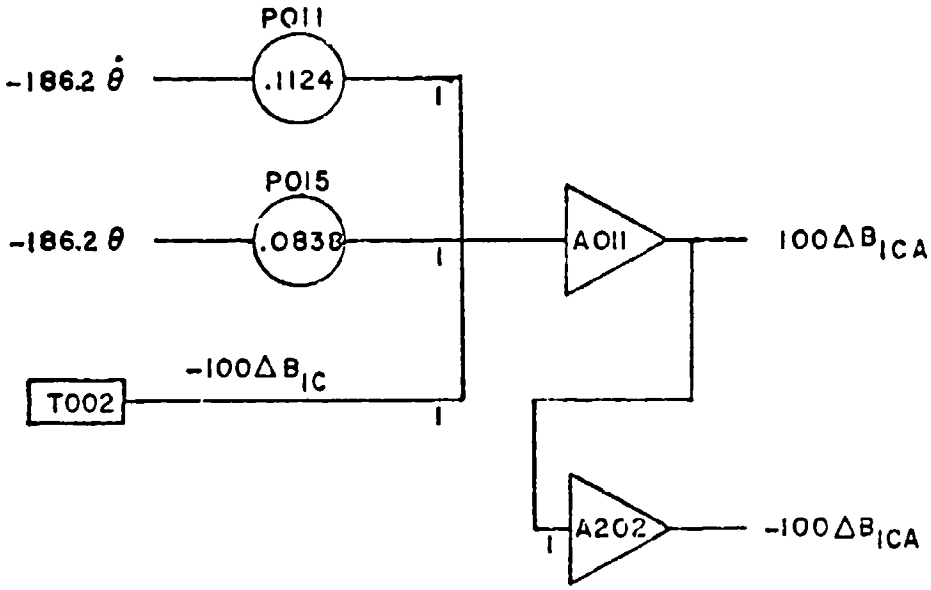
PITCH MOMENT EQUATION



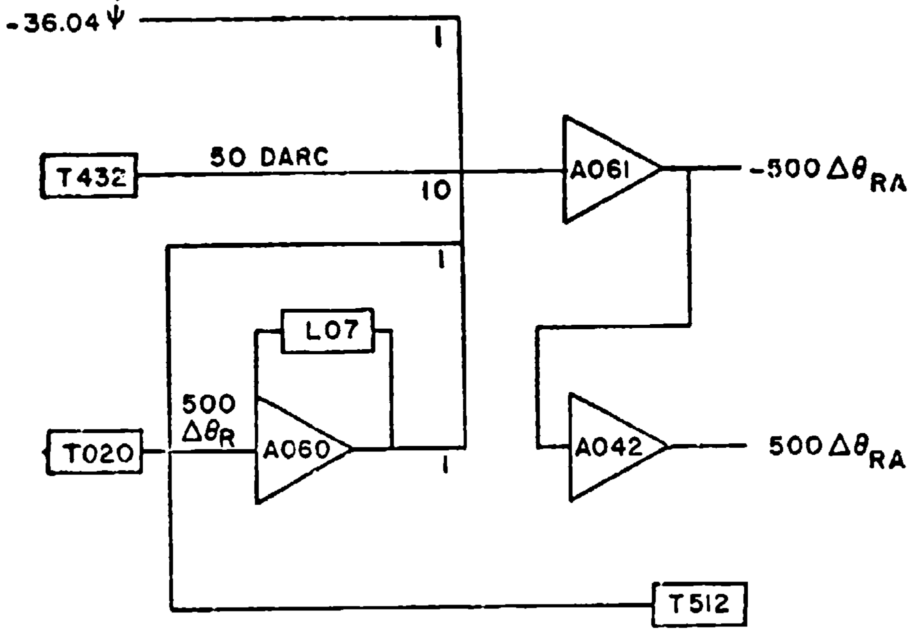
YAW MOMENT EQUATION



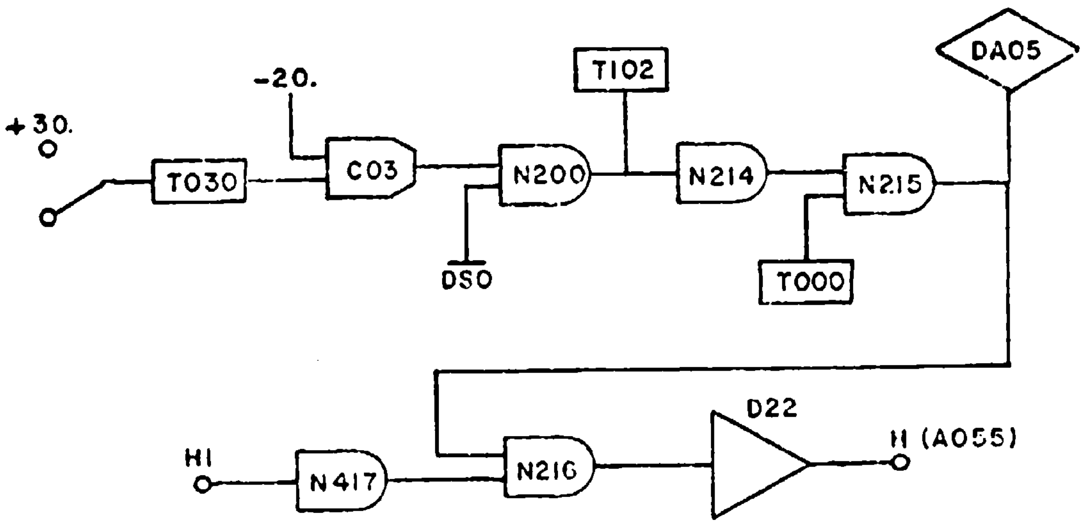
ROLL AUGMENTATION



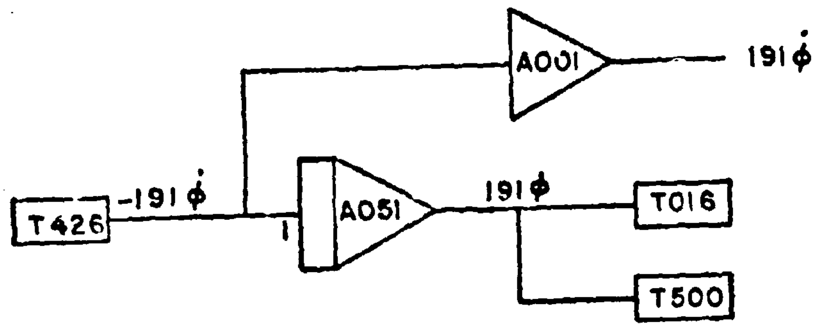
PITCH AUGMENTATION



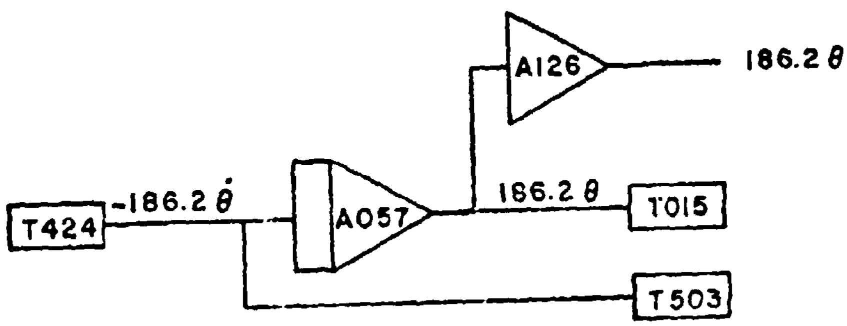
YAW AUGMENTATION



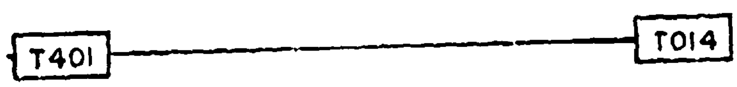
COORDINATED TURN



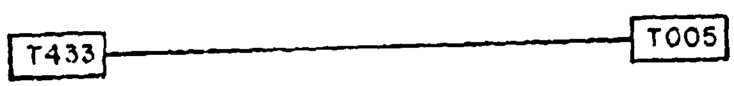
ROLL ANGLE



PITCH ANGLE

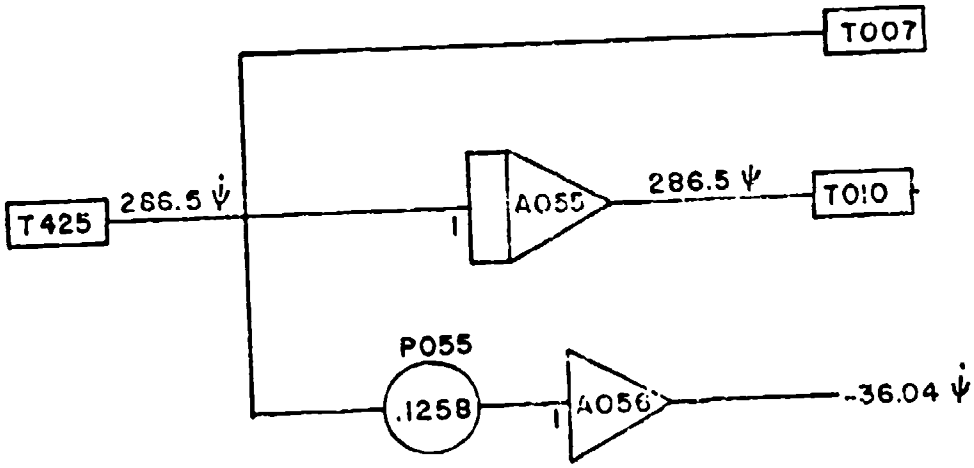


RADAR ALTIMETER



AIRSPEED INDICATOR

VERTICAL SPEED INDICATOR


T007
T425
286.5 $\dot{\psi}$
A055
1
286.5 ψ
T010
P055
.1258
A056
1
-36.04 $\dot{\psi}$

YAW ANGLE

APPENDIX D

PILOT QUALIFICATIONS AND RATINGS

PILOT	FLIGHT HOURS	HELICOPTER HOURS	INSTRUMENT HOURS	CONVENTIONAL INSTRUMENTS	INTEGRATED DISPLAY
A	1100	980	205	8	8
B	2512	2313	296	7	6
C	1000	800	150	10	9
D	1130	975	215	9	8
E	1450	1050	240	8	8

LIST OF REFERENCES

1. U.S. Army Aviation Systems Command, Minutes of Helicopter Instrument Flight Conference, Edwards AFB, California, 30-31 January 1973.

2. Hoxie, Stephen Sanders, The Implementation of a Fixed Base Helicopter Simulation in the Investigation of an Automatic Scan System, Engineers Thesis, Naval Postgraduate School, Monterey, California, March 1974.

3. NAVAIR 01-230HLC-1, NATOPS Flight Manual SH-3A/G Helicopters, 15 June 1969.

4. Electrical Engineering Computer Laboratory, Manual for the Fortran User, Naval Postgraduate School, Monterey, California, 10 February 1972.

5. Sweeney, C. J., Conversion of AC-11B Instrument Flight Trainer into a Variable Stability Flight Simulator, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1968.

6. Huckemeyer, M. R., Incorporation of the Six Dimensional Linearized Equations of Motion with a Variable Stability Flight Simulator, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1971.

7. Etkin, Bernard, Dynamics of Flight, p. 102-103, Wiley, 1959.

END